National Library of Canada
Acquisitions and Bibliographic Services Branch
395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada
Direction des acquisitions et des services bibliographiques
395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file *Votre référence*

Our file *Notre référence*

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

University of Alberta

The Development and Validation of a Computerized IPP Management System through Prototyping

by

David Lyle Wodelet

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Education.

in

Instructional Technology

Department of Adult Career and Technology

Edmonton, Alberta

Fall 1993

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-88320-0

Canada

University of Alberta

Release Form

Name of author: **David Lyle Wodelet**

Title of thesis: **The Development and Validation of a Computerized IPP Management System through Prototyping**

Degree: **Master of Education**

Year this degree was granted: **1993**

Permission is hereby granted to the University of Alberta library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in this thesis, and except as herein before provided neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

D L Wodelet.

33 Chelsea Way
Sherwood Park, Alberta
Canada, T8H 1E6

August 27, 1993

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **The Development and Validation of a Computerized IPP Management System through Prototyping** submitted by **David Lyle Wodelet** in partial fulfillment of the requirements for the degree of **Master of Education** in **Instructional Technology**.

Dr. M. Szabo

Dr. T. C. Montgomerie

Dr. D. J. Collett

July 28, 1993

Abstract

It is projected that the teacher's role in individualized instruction will increase from 5% in 1990 to 50% by the year 2000. However, if the move towards individualized instruction is to be successful, then it is important that its introduction be matched with the introduction of automated (computerized) management systems to offset the increased workload for the teacher. The purpose of this study was to design, develop, implement and validate an automated/computerized system for the management of students in individualized learning environments. It was designed to answer the following research questions:

1) What features/components should such a system contain?
2) What is involved with the development and implementation of such a system?
3) What are the users' views and perceptions after implementation of such a system?

The instruments and techniques used to collect the data for this study were interviews, document reviews, and personal notes. The methodology used to develop the computerized system was based on prototyping. The implementation and data gathering was carried out in one public school system in Alberta, Canada.

Table of Contents

Chapter	Page
I. Overview of the Problem	1
Introduction	1
Problem Statement and Research Questions	2
Statement of Significance	2
Delimitations	3
Limitations	3
Definition of Terms	4
II. Review of the Literature	8
Introduction	8
Individualized Instruction: In Perspective	8
Individualized Instruction: The Challenge	9
Meeting the Challenge with Technology	10
Management of Individualized Learning	12
Computer Managed Instruction	12
Computerized Management of IPPs	13
The Need for a Systemic/System Approach	15
Concluding Remarks	16
III. Methodology	18
Introduction	18
Instruments and Techniques	18
System Features/Components	18

System Development and Implementation 19
Post-Implementation Views and Perceptions 19
The Interview Process 20
The Sample 20
Enhanced Learning Assistance (ELA) 21
Student Characteristics 21
Program Characteristics 22
Educable Mentally Handicapped (EMH) 22
Student Characteristics 22
Program Characteristics 22
Trainable Mentally Handicapped (TMH) 23
Student Characteristics 23
Program Characteristics 23
Ethical Concerns and Considerations 23
IV. System Design, Development and Implementation 25
Introduction 25
Description of the Research Environment 25
System Features/Components 26
Review of Existing Systems 26
IEPWorks 27
IEP Database 28
List of Features/Components 28
Interpreting the Data 33
Interrelationships of Features 33
Prototyping versus System Reviews 34
System Development and Implementation 35
The Prototyping Process 35

The Development Language 36
Description of the Developed System 37
Overview 37
Creating a New IPP 39
Working With Objectives 39
Viewing and Printing Objectives 41
Add Repeating 41
Inserting/Adding Objectives 42
Changing and Deleting Objectives 42
The Objective Databases 43
Selecting an IPP Format 43
Customizing the IPP 49
Changing Column Headings 49
Changing Column Widths 49
Inserting New Columns 50
Changing How the Program Works 51
Development and Implementation Issues 52
Implementation of Features 52
Using a High-level Language 52
User / Developer Communication 53
Program Updates 53
Backward Compatibility 54
Program Version Management 54
Use of Templates 54
Program Automation 55
Safeguarding the System 55
Safeguarding the Student's Data 55

Training and User Support 55
Modular System Design 56
Shared Variable Name Space 56
Variable Creation and Destruction 57
Subroutine Arguments 57
Program Layout 57
User Input Layout 58
Source Code Debugging 59
V. Post-Implementation Views and Perceptions 60
Introduction 60
Interview Summary 60
Who is Using the System 62
How the System was Rated 64
How the System is Being Used 66
User Training and Support 71
Future Use of the System 73
Suggestions for Improving the System 74
VI. Conclusion and Recommendations 79
Overview 79
Existing IPP Management Systems 79
Prototyping Versus System Reviews 80
Excel as a Prototyping Language 80
User / Developer Communication 81
Computer Literacy 81
Training 82
Computer Access 82
Recommendations for Future Development 83

Bibliography 85

Appendices 89

A. Consent Form 90

B. Pre-Interview / Contact Guide 92

C. Interview Guide: Determining System Features 93

D. Interview Guide: Users' Views and Perceptions 95

E. Program Listing 97

F. User Manual 108

G. IEPWorks: Sample IPP 132

H. IEP Database: Sample IPP 140

I. IPP Information Form 147

Curriculum Vita 152

Diskette Enclosure 160

List of Tables

Table		Page
Table 1:	Number of Staff and Students in Program	26
Table 2:	List of Features / Components	29
Table 3:	Staff Interviewed for Post-Implementation Views and Perceptions	61
Table 4:	Total Responses per Question	61
Table 5:	Question #1 — Did you use the IPP computerized management system?	63
Table 6:	Question #1 — Comment Summary of Reasons For Not Using the System	63
Table 7:	Question #16 — How would you rate your comfort level in working with computers?	64
Table 8:	Question #1[illegible] — How would you rate the system overall?	65
Table 9:	Question #9 — Once you learned how to use the computerized IPP system did it save time?	65
Table 10:	Question #2 — When did you start using the IPP system?	67
Table 11:	Question #3 — How often do you use the actual computerized system?	68
Table 12:	Question #4 — How often do you use the data sheets provided by the system?	68
Table 13:	Question #13 — Of the students you need to write IPPs for, approximately what percentage these students did you put on the system?	68
Table 14:	Question #14 — How many students do you write IPPs for?	69
Table 15:	Question #15 — On the average, how many IPP objectives do you maintain on a student?	69
Table 16:	Question #17 — Did you need to change the objectives provided in the database?	70
Table 17:	Question #17 — Frequency of Changes to Database Objectives	70

Table 18: Question #17 — Degree of Changes to Database Objectives 70

Table 19: Question #6 — Did you attend the provided training sessions? 71

Table 20: Question #6 — Comment Summary of Reasons For Not Attending Training Sessions 71

Table 21: Question #7 — Was adequate training provided? 72

Table 22: Question #7 — Comment Summary of Reasons Why Training was Inadequate 72

Table 23: Question #10 — Was adequate support provided when you had problems? 72

Table 24: Question #8 — Did you use the provided user manual? 73

Table 25: Question #12 — Would you recommend continuing use of the system? 73

Table 26: Question #12 — Comment Summary of Reasons for Not Continuing Use of the System 74

Table 27: Question #5 — Do you anticipate your future use of the system to increase, decrease or stay about the same? 74

Table 28: Question #18 — Should any changes be made to the system in order to make it more useful for you? 76

Table 29: Question #19 — Should more of the IPP process be computerized? 76

Table 30: Question #20 — Are there any additional features/components, not already covered, you deem important for an ideal/future system? 76

Table 31: Question #18, 19, 20 — Comment Summary of Ways to Improve the System 77

Table 32: Administrative Computers per School 78

Table 33: IPP Diskette Contents 161

List of Figures

Figure **Page**

Figure 1: Sample Screen (Check List Format) 37

Figure 2: Diskette Contents 38

Figure 3: Creating a New IPP 39

Figure 4: Inserting Objectives 40

Figure 5: Contents of the "IPP Data Files" Folder 40

Figure 6: Sample Database File Window (EMH) 41

Figure 7: Changing the IPP Format 44

Figure 8: Sample Check List Format 45

Figure 9: Sample Report Card Format 46

Figure 10: Sample Achievement Format 47

Figure 11: Sample Program Plan Format 48

Figure 12: Changing Column Widths 49

Figure 13: Inserting a New Column 51

Figure 14: Program Source Code Layout 58

Figure 15: Dialog Box Data Table Layout 58

Figure 16: Macro Error Message 59

List of Symbols
Nomenclature and Abbreviations

XXX School District — The code "XXX" is used in this thesis to identify the school district at which the study was carried out. This code was used to maintain confidentiality.

Chapter I

Overview of the Problem

Introduction

Advancements in instructional technology, such as Computer Based Instruction (CBI), are making it increasingly more feasible for students to progress at their own rates and tackle their own unique objectives. However, the benefits that individualized instruction brings to the student may never be realized unless the new sets of problems it creates for the teacher are overcome (David, 1991; Alberta Education, 1987; Baker, 1981; Davies, 1973).

> One of the major deterrents to offering a truly individualized program is the formidable amount of diagnostic work, record-keeping and administration involved for teachers who generally have too little time to undertake this work (Alberta Education, 1987, p. 17).

> Modern instructional strategies are becoming more complex and thus harder to manage from an instructional and administrative point of views. Individualized rate of progress schemes, in particular, pose severe management loads. ... Experience has shown that it is not feasible economically to implement individualized instruction without supporting it with CMI [Computer Managed Instruction — an automated support system for managing instruction]. The enormous clerical and monitoring burdens simply overwhelms existing instructional management paradigms when more that a handful of students are involved. (Baker, 1981, pp. 23-25)

A review of the literature indicates that there has been little progress in addressing these problems over the past few years. Furthermore, technological advancements over the last few years have only served to exacerbate the situation (David, 1991). These advancements have increased the complexity of the teacher's job by making more complicated, though more effective, approaches to teaching possible.

> In fact, introducing technology into schools as currently organized vastly increases the complexity of teachers' jobs because it makes possible more complex — though more effective — approaches to teaching. ... Active [individualized] learning environments, by increasing student movement

> and communication, also pose organizational, planning, management and evaluation challenges to even the strongest and most innovative teachers (David, 1991, p. 79).

It is projected that the teacher's role in individualized instruction will increase from 5% in 1990 to 50% by the year 2000 (Alberta Education, 1987, p. 17). This projected increase will also be paired with a corresponding increase/reliance on technology to help deliver this form of instruction. However, if the move towards individualized instruction is to be successful, then its introduction should be matched with the introduction of automated (computerized) management systems to offset the increased workload for the teacher. The importance of this systemic/system approach to the application of technology has been stressed frequently in the literature (La Follette, 1992; Reiser & Salisbury, 1991; David, 1991; Dick & Carey, 1990; Davies, 1973; Lange, 1967). This synergistic application of technology would be necessary to give teachers more time to concentrate on their more crucial roles of planning learning experiences and guiding, motivating, and tutoring learners.

Problem Statement and Research Questions

The purpose of this study was to design, develop, implement and validate an automated/computerized system for the management of students in individualized learning environments. This led to the following research questions:

1) What features/components should such a system contain?
2) What is involved with the development and implementation of such a system?
3) What are the users' views and perceptions after implementation of such a system?

Statement of Significance

There is currently a high level of concern in both education and industry about how our students and employees measure up with those in the rest of the world. In addition, our rapidly changing technologically-based society has created a need for life-long learning to enable employees to keep pace with their rapidly changing employment requirements. Both these factors, in combination with the dramatic achievement gains possible with individualized instruction (Bloom 1984), will result in an increased movement towards this form of instruction in the future (Gentry & Csete 1991). Alberta Education (1987), in their *Visions 2000: A Vision of Educational Technology in Alberta by the Year 2000*, projected that the teacher's role in individualized instruction will increase from 5% in 1990 to 50% by the year 2000. Due to the high cost of delivery of this form of instruction through the use of live teachers, more pressure will be placed on technology to help solve the growing need (Clark & Sugrue, 1988). However, care must be taken when introducing such isolated technological change, otherwise failure may result due to unwanted changes in other, but related, parts of the system (Davies, 1973; Lange, 1967). To avoid this, many authors have stressed the importance of a systemic/system approach

for the successful implementation of new technology (La Follette, 1992; David, 1991; Reiser & Salisbury, 1991; Dick & Carey, 1990; Davies, 1973; Lange, 1967).

Therefore, this study gains its significance from the anticipated shift to individualized learning environments and the corresponding need for complementary systems to address problems caused by this shift of instructional modes. If these problems aren't identified and solved, the move towards individualized instruction is likely be severely impeded — a systemic/system approach is needed. This research will help provide future developers of these complementary systems the information they need to design useful and effective automated management systems for both education and industry.

Delimitations

This study was delimited in the following ways. These delimitations were necessary in order to reduce the scope of this topic to a manageable size and to permit a more focused and rigorous examination of the research questions.

- The individualized learning environment under investigation related to Individualized Program Plans (IPPs). This area of individualized instruction was, at the time of the study, the most active with more wide spread use. Other individualized learning environments, such as those involving CAI, did not have the same degree of entrenchment in the marketplace.
- Not all the components/features deemed important/useful for such a system were implemented in the developed system due to resource constraints (time, money, available machine architectures, available processing power, etc.).
- The document review of existing systems was limited to a purposive sample from the automated systems currently available and in use in the marketplace at the time of the study.
- The teachers and staff involved in the study came from a single large school district in Alberta.
- The individualized learning environment under investigation was restricted to an Individualized Program implemented in the special education area within an ECS to grade 12 school system.

Limitations

This study was subject to a series of limitations. Efforts were made to keep the limitations to a minimum. However, as in all studies, some limitations are inevitable.

- All teachers and staff interviewed came from a special education background. Although Individualized Program Plans (IPPs) could be used for non-special students this is currently not being done to any great degree by school boards within the province nor by the school board under study. Data for this study came from

three special education areas: Trainable Mentally Handicapped (TMH), Educable Mentally Handicapped (EMH) and Enhanced Learning Assistance (ELA).

- The teachers and staff participated in this study on a voluntary basis. Since some individuals did not agree to participate, their views, comments and perceptions could not be gathered.
- Software suppliers may have not fully disclosed all components/features they perceived important for the management of IPPs. In order to maintain a competitive advantage they may have not wanted competitors to know specific internal details/features of their product.
- The components/features considered important are reflective of the social/political context at the time of the study. What is considered important often changes as school, government and/or industry requirements and needs change.
- Limitations due to response effects — the tendency of respondents to give inaccurate or incorrect answers — are always possible when conducting interviews. To help reduce this type of error, suggestions given by Borg & Gall (1989) and Sudman & Bradburn (1983) were used. The respondents were frequently reassured that confidentially of their comments would be strictly maintained. In addition, all interviews were held in an informal atmosphere and in a relaxed manner (over coffee; interviewer and respondents sat with each other rather than across an office desk; interviewer maintained casual clothing so it wasn't perceived as a central office / official business visit).

Definition of Terms

This section defines a number of terms to assist the reader. The fields of educational and instructional technology are dynamic and, in many ways, still emerging fields. In their relatively short periods of evolution, many terms used in these fields have taken on a surprisingly wide range of meanings (Gentry, 1991). As these fields mature so will the terms they use — congealing into more common, widely accepted, definitions. This section attempts to define the terms used in this study to provide a common base for understanding and discussion. These definitions are only as detailed as necessary in order to meet that goal and are not intended to be rigorous definitions covering all the nuances and variations of their use.

CAI — acronym for Computer Assisted Instruction. Refer to this definition for more detail.

CBE — acronym for Competency Based Education. Refer to this definition for more detail.

CBI — acronym for Computer Based Instruction. Refer to this definition for more detail.

CMI — acronym for Computer Managed Instruction. Refer to this definition for more detail.

CML — acronym for Computer Managed Learning. Refer to the definition of Computer Managed Instruction for more detail.

Competency Based Education (CBE) — a form of training or instruction which requires the learner to attain a preset level of competency in the area being taught before they are allowed to move on to the next area or objective. It is an educational system that emphasizes the specification, learning, and demonstration of those competencies (knowledge, skills, behavior) that are of central importance to a given task, activity or career (E.R.I.C. Thesaurus).

Computer Assisted Instruction (CAI) — the use of computing technology to present instructional material in accordance with individual learner needs. This consists of three basic processes in a variety of combinations: 1) presents and structures information, 2) accepts and evaluates student's responses, and 3) routes the student through one of several instructional paths. This is one of the two components which make up computer based instruction (CBI).

Computer Based Instruction (CBI) — the use of computing technology to present instructional material (CAI component) as well as to monitor/manage the learning process (CMI component).

Computer Based Training (CBT) — the delivery of training using computing technology as the medium of instruction. For the purposes of this study, this term can be considered synonymous with Computer Based Instruction.

Computer Managed Instruction (CMI) — also known as Computer Managed Learning (CML). The use of computing technology to maintain and analyze data on learner performance and instructional progress as an aid to teachers in selecting and managing learning activities (E.R.I.C. Thesaurus). Ideally CMI should play both a diagnostic and prescriptive role in the management of instruction. CMI is one of the two components which make up computer based instruction (CBI).

Educable Mentally Handicapped (EMH) — students in this program are described as mildly mentally handicapped with a significant academic lag. Their IQ generally falls between 50 and 80. In terms of grouping, EMH students are the next ability level above TMH (Trainable Mentally Handicapped) students and the next ability level below ELA (Enhanced Learning Assistance) students. A more detailed discussion of these students is provided in the **Methodology** chapter (XXX School District, 1992).

Educational Technology — the combination of instructional, learning development, managerial and other technologies as applied to the solution of educational problems (Gentry, 1991). One major area of this field of study is that of *instructional technology*.

ELA — acronym for Enhanced Learning Assistance. Refer to this definition for more detail.

EMH — acronym for Educable Mentally Handicapped. Refer to this definition for more detail.

Enhanced Learning Assistance (ELA) — students in this program are described as having average ability but with severe academic difficulties and specific learning disabilities. Their IQ is generally from 80-90 and above. In terms of grouping, ELA students are the next ability level above EMH (Educable Mentally Handicapped) students. A more detailed discussion of these

students is provided in the **Methodology** chapter (XXX School District, 1992).

Fourth Generation Language (4th GL, 4GL) — a high level computer programming language which has the attributes of being user-friendly, usable by non-programmers, non-procedural, designed for easy debugging, and is capable of making intelligent decisions about what the user wants (where possible). There seems to be no rigid, universally accepted, formal definition of a 4th GL probably because of the continual stage of change/evolution computing languages are in. It seems that what is a 4th GL is more often defined by ruling out what it is not — i.e. showing it is not a 3rd GL, 2nd GL or 1st GL. (Martin, 1984, 1989).

FTE — acronym for Full Time Equivalent. Allocating a 0.5 FTE to a program means allocating staff resources equal to half the time of a full time teacher. These resources could be provided by a single teacher working half time (0.5) or by two teachers working quarter time (0.25 x 2), etc. This acronym is used in some of the documentation provided on the XXX School District.

IEP — acronym for Individualized Educational Plan. This can be considered synonymous with Individualized Program Plan (IPP). Refer to this definition for more detail.

IIP — acronym for Individualized Instructional Plan. This can be considered synonymous with Individualized Program Plan (IPP). Refer to this definition for more detail.

Individual Program Plan (IPP) — a program plan adapted to meet the individualized needs of a learner within a group. This can be considered synonymous with other terms such as Individualized Educational Plan or Individualized Instructional Plan. This is an educational program for individual students, each geared to the particular student's needs and conducted in accordance with a written plan agreed on between the student (and/or his/her parents) and school officials. IPPs were originally conceived for use in educating handicapped children but have gradually evolved to include other special needs and non-special needs students.

Individualized Instruction — instruction tailored to the individual/specific needs and pace of a learner. This encompasses both one-to-one tutoring as well as such automated instruction as CAI.

Individualized Learning Environments — an environment or setting designed or tailored for the individual/specific needs and pace of a learner. An environment designed to facilitate the delivery of *individualized instruction.*

Instructional Technology — the systemic and systematic application of strategies and techniques derived from behavior and physical sciences concepts and other knowledge to the solution of instructional problems (Gentry, 1991). This field of study is considered a subset of the larger field of *educational technology.*

IPP — acronym for Individualized Program Plan. Refer to this definition for more detail.

Prototyping — a systems development methodology where the design and development phases of the traditional software development are shortened. In prototyping, initial user requirements are gathered, incorporated/developed into a prototype, and then the resulting prototype is "quickly" returned to the user for evaluation and implementation. Then the next set of design requirements are gathered to begin the cycle again. Prototyping usually involves the use of a high level (fourth generation) language (Martin, 1984, 1989).

Systemic/System Approach — essentially the Gestalt concept that the whole is more than a simple summation of its constituent parts. The terms *systemic* or *system approach* can be considered synonymous terms; they are frequently used interchangeably in the literature. This approach, when applied to technology, involves viewing the application of technology in a more holistic as opposed to a piece-meal manner. It involves considering the effect some application of technology will have on other parts of the system (Dick & Carey, 1990; Davies 1973).

TMH — acronym for Trainable Mentally Handicapped. Refer to this definition for more detail.

Trainable Mentally Handicapped (TMH) — students in this program are described as moderately handicapped with an IQ that generally falls between 30 and 50. In terms of grouping, TMH students are the next ability level below EMH (Educable Mentally Handicapped) students. A more detailed discussion of these students is provided in the **Methodology** chapter (XXX School District, 1992).

Chapter II

Review of the Literature

Introduction

The purpose of this literature review was to identify the progress made relative to the major and minor concepts related to individualized instruction and instructional techniques. In addition to the print based indices, a number of electronic databases were used to uncover and compile relevant works in the area. These proved by far to be the most useful. The electronic databases used included those from the Educational Resources Information Center (ERIC), the University of Alberta's On-line Catalog, and BRS Information Technologies. In most cases the database searches were limited to a five year time window (1987 or greater). However, in cases where insufficient material was found the search was expanded to approximately a ten year time window (1982 or greater). In a few cases, works earlier than this have been cited in this review due to their historical significance.

Individualized Instruction: In Perspective

Individualized instruction is instruction tailored to the individual needs and pace of the learner. This implies a highly interactive learning environment centered around the learner. This form of instruction can take many forms ranging from highly personal one-to-one tutoring, to semi-advanced programmed learning, to highly automated computer based instruction (CBI). One-to-one tutoring is as old as humankind itself and has been shown to be an extremely effective method of instruction (Bloom, 1984; Gentry & Csete 1991). Computer based instruction (CBI) is a fairly recent addition to this age-old form of instruction. To date, its effectiveness has not matched that of one-to-one tutoring but it has been shown to produce significant achievement gains by a number of meta-analytic studies (Lee, 1990; Niemiec, Samson, Weinstein & Walberg, 1987; Bangert-Drowns, Kulik & Kulik, 1985; Kulik, Kulik & Cohen, 1980).

The relationship that individualized instruction has with instructional technology cannot be fully appreciated unless one looks at some evolutionary trends or milestones in

instructional technology. One of these evolutionary trends was the paradigm shift in instructional media research from behavioral to cognitive theories in the past decade (Clark & Salomon, 1986). A behavioral theory of learning focuses on environmental causes of changes in behavior/learning without reference to the mental processes mediating such changes. In contrast, a cognitive theory of learning views learning as a constructive process, with the learner actively engaged in the process of integrating new knowledge with old. In the cognitive paradigm, it's the learner's attributes — past experiences, prior knowledge, preferences, intellectual abilities, interests and attitudes — which determines the effectiveness of the instructional process on learning. The importance of learner attributes and past experiences in the learning process is supported by a number of active areas in cognitive learning such as experiential learning (Kolb, 1984), generative learning (Wittrock, 1974), and attribute-by-treatment interaction (Dwyer, 1978). This paradigm shift has caused a renewed[1] realization of the importance of individualized instruction in the learning process. When the learner's attributes play such an important role in determining whether learning results from instruction, then the *most effective* learning environment is one where these attributes are taken into account on an individualized basis. This is precisely the type of environment individualized instruction strives to create.

Individualized Instruction: The Challenge

Based on the cognitive theories of learning, one would expect individualized instruction to be more effective than conventional group instruction because it is tailored to meet the needs of each individual learner. Bloom (1984) has shown this to indeed be the case. He found that students who were taught the same content, in a traditional manner, with the only difference between experimental and control groups being a tutoring component, resulted in the average tutored student learning at 2 sigma (standard deviations) above the average student in the control group. In other words, the average tutored student was in approximately the 95th percentile of the control group. However, because of the prohibitive costs of tutoring, he doesn't see this as a solution to raising the performance level in schools. From this work he formulated the following question which has come to be known as the "**2 Sigma problem**":

> Can researchers and teachers devise teaching-learning conditions that will enable the majority of students under *group instruction* to attain levels of achievement that can presently be reached only under good tutoring conditions? (pp. 4-5)

Whether this can be accomplished through improving conventional group instruction is uncertain but seems unlikely (Gentry & Csete, 1991). Therefore, the question instructional technologists are asking themselves instead is:

---

[1]The importance of a student's past experiences in relation to the learning process can be traced as far back as Dewey (1916) — a forerunner in stressing the linkage between cognitive processes and concrete experiences.

> Are there other economically feasible ways of meeting this standard through use of instructional technology and media in our schools and in industry?

Meeting the Challenge with Technology

Computers, the world's first generalized tool (Palfreman & Apsell, 1992), and their associated media, seem to be the best candidates so far to even come close to being able to meet the challenge posed by the 2 Sigma problem — to attain the same levels of achievement that can presently be reached only under good one-to-one tutoring conditions with "live" teachers.

> ...computers and videodisc media may provide the constant interaction that individualized instruction requires but has only been previously available from expensive live teachers (Clark & Sugrue, 1988, p. 34).

To date, the effectiveness of CAI has not matched that of one-to-one tutoring but it has been shown to produce significant achievement gains by a number of meta-analytic studies (Lee, 1990; Niemiec, Samson, Weinstein & Walberg, 1987; Bangert-Drowns, Kulik & Kulik, 1985; Kulik, Kulik & Cohen, 1980). The results of these studies have shown that, on average, achievement scores were raised by approximately 0.30 to 0.45 standard deviation units. This modest achievement gain is encouraging but certainly no where near the 2 standard deviations reported by Bloom (1984) when using live tutors.

It is not clear why only modest achievement gains have been seen so far. It has been suggested that neither CBI nor any other computer based delivery system will result in the gains in achievement levels being strived for.

> Research on computer based instruction (CBI) has been supportive of its effectiveness (Kulik, Bangert & Williams, 1983), yet it is unlikely that CBI or any other computer-based delivery system will produce results necessary to proclaim it "better" than traditional methods (Morrison, Ross & O'Dell, 1991, p. 188).

This lack of achievement gains can best be explained by examining some convincing arguments which contend that neither computers, nor any other media, are directly responsible for achievement gains in learning. This lack of effectiveness can be explained by conceptualizing media (which computers are but one type) as delivery vehicles for instruction and not as variables which directly influence learning. Since these delivery vehicles do not directly influence learning, no appreciable achievement gains would be expected to come from using them.

> The conclusion that media do not influence learning directly can be summed up in an analogy: In instruction, media serve a function similar to the different forms in which prescriptions medicines are delivered. One would not claim that a tablet or a liquid suspension of a drug altered the effects of the drug on human biological functions (except to make it more or less efficient). Nor is it important, except for efficiency purposes, whether a drug is administered by the medium of injection or by oral ingestion. It is

> the prescription compound that influences biology, not the medium of delivery (Clark & Sugrue, 1988, p24-25).

It has been argued that most of the achievement gains seen are due to either uncontrolled research variables, novelty effects of the new medium, learner's views or perceptions about the media and/or the organizational context surrounding its introduction — not the media itself (Clark & Sugrue, 1988; Clark & Salomon, 1986; Clark 1983).

> The students' increased attention sometimes results in an increased effort or persistence, which yields achievement gains. If attentiveness is due to a novelty effect, these gains tend to diminish as students become more familiar with the new medium. This was the case in reviews of computer-assisted instruction at the secondary school level (grades 6 to 12). ... Even in the few cases where dramatic changes in achievement or ability were found to result from the introduction of new medium such as television or computers, it was not the medium per se that caused the change but rather the curricular reform that accompanied the new medium (Clark & Sugrue, 1988, pp. 25-26).

The argument that computers, as delivery vehicles for instruction, do not directly influence learning appear valid. However, in the strictest sense, human teachers could be considered delivery vehicles as well. In that case, how can the achievement gains seen by Bloom (1984) be explained? The reason lies in the fact that teachers are more than mere "delivery vehicles". They are able to interactively adapt their method of instruction to best match the specific needs of the learner depending on the attributes (past experiences, attitudes, etc.) they possess. In the case of group instruction this match of instruction to learner attributes wouldn't be as specific as in individual instruction, hence the lower achievement gains. However, group instruction is still effective due to an overlap of learner attributes in the group — individuals in the learning group are usually products of similar environments (same geographical area, similar socio-economic class, etc.).

However, by the same reasoning, computers, as generalized machines, can be more than mere delivery vehicles as well. As a mere delivery vehicle they probably will be no more effective than any other medium — that explains the poor achievement results of previous studies. However, as intelligent tools it is possible for them to interactively adapt their method of instruction to best match the specific needs of the learner — just as live tutors would do. It appears that any great strides to be made by this technology will be more dependent on advances in software and delivery methods rather than advances in hardware.

> The future of computer-based delivery systems appears to belong to the designer of the instructional strategy as opposed to the computer hardware (Morrison, Ross & O'Dell, 1991, p. 188).

The question is whether or not the field of instructional technology, in the area of instructional design, is currently mature enough to enable designers to make the changes necessary to effectuate dramatic achievement gains. The poor performance of existing CBI suggests that this is not the case. The same achievement gains that exist for one-to-one tutoring will not be realized until future instructional systems have the same intelligence and/or skill that the human tutor does in matching the learner's attributes to the optimal mode of instruction. The best hope seems to lie in future advances in knowledge based systems, tightly integrated multi-media systems, and automated development tools (Morrison, Ross & O'Dell, 1991). Advances in these areas may help instructional designers to create systems which are more effective and adaptable to

different types of learners. This would provide not only the optimal instructional modes, but challenge the student to develop proficiencies in other alternative modes[2]. For these systems to be as effective, they would probably need to emphasize program control rather than learner control for some learners — just as real tutors do. Research suggests that learner control options may not always be effective when individualizing instruction (Carrier, Davidson & Williams, 1985; Tennyson, 1980). It appears that most learners aren't capable of correctly assessing their current level of knowledge and, when instruction is self-paced, may end up terminating their instruction prematurely before they have mastered the objectives (Tennyson, 1981).

> Many students, especially low achievers, lack the knowledge and motivation to make appropriate decisions regarding such conditions as pacing,... sequencing content,... use of learning aids,... and amount of practice (Ross & Morrison, 1989, p. 28).

Intelligent program control could probably only be achieved through use of up to date databases of student profiles. These profile databases would probably use data collected on how successful prior types of instruction and media had been in order to help the smart systems make effective predictions as to the optimal instructional path to take.

Management of Individualized Learning

Computer Managed Instruction

The management of instruction is a key element of computer related research in education. The computerized management of the instructional process has come to be known as Computer Managed Instruction (CMI).

> CMI has evolved into an information system designed to facilitate the management of instruction and individualized instruction in particular. It provides the automated data collection, data processing, and reporting capability needed to cope with the managerial demands of individualized instruction. (Baker, 1981, p. 25)

The history of CMI can be traced as far back as the first commercial system designed specifically for the delivery of Computer Assisted Instruction (CAI) — the IBM 1500. Shortly after its introduction in 1968, developers began to realize the importance of a

---

[2] In learning style approaches based strictly on experiential learning, you don't need to diagnose a student's learning style or accommodate their perceptual and processing preferences. This is because, with the experiential learning approach, all students should participate in activities in all four adaptive learning modes (Kolb, 1984). However, if you could determine the optimal mode for a student, the "smart" system could manage the instruction in such a way to ensure the correct amount of compensation/remedial help is given in the non-preferred modes for the student.

management component to complement their development of the instructional (CAI) component.

> As courses [on the IBM-1500] were developed during the early 1970s, authors began to realize that, while Coursewriter II [the CAI development language on the IBM-1500] provided good instructional capabilities, there was no predefined ability to manage the testing and routing of students through the course. Each author needed to build such computer-managed instruction (CMI) capabilities into the CAI code. (Szabo & Montgomerie, 1992, p. 114)

The need for a management component culminated with the integration of CMI and CAI into a single authoring system with the advent of the CDC PLATO system in the early 80s. Szabo and Montgomerie (1992) in their paper on *Two Decades of Research on Computer-Managed Instruction* provide an informative overview of the history and evolution of CMI and review a number of key studies in the area.

CMI is often broken down and conceptualized as an interrelated system of parts. The number of parts vary depending on the author, but often includes such components as curriculum plans, diagnostic evaluations and learning prescriptions, instructional strategies, reporting, module scheduling, instructional/historical student record maintenance and/or student counseling/feedback regarding optimal learning strategies (Baker, 1981; Hansen, 1970 cited in Szabo & Montgomerie, 1992). Although there is a broad spectrum of components which could be considered part of a CMI system, the core elements are often considered diagnosis and prescription (Baker, 1981; Szabo & Montgomerie, 1992). The diagnostic component diagnosis the student's ability or performance level and provides the necessary information to the prescriptive component which prescribes the most appropriate form of instruction (ideally taking into account the student's past instructional history as well as all curricular requirements).

It is important to note that throughout its history CMI has had very strong ties with CAI. Together, these two components comprise what has come to be known as Computer Based Instruction (CBI). These ties have been so strong that this probably contributed to the reason why "the research on pure CMI is thin" (Szabo & Montgomerie, 1992, p. 127). However, because CBI does contain a CMI component, if one looks at the overall effect indicated by several similar CBI studies, CMI appears to be an effective and important part of the instructional process.

Computerized Management of IPPs

The preceding discussion concentrated on the management of individualized instruction as delivered primarily in conjunction with a CAI component. However, another mechanism used for the delivery of individualized learning, and the main focus of this research, involves Individualized Program Plans (IPPs). Unfortunately, research on the computerized management of IPPs appears to be even more "thin" than the research is on pure CMI (C.A. McClure, personal communication, May 31, 1993).

> While teachers are attracted to the time and cost savings and enhanced attitudes realized with the generation of computerized individual education plans [also called IPPs], critics lament the dearth of evidence to support

> claims for procedural compliance, substantive content and program implementation quality (Reynolds, 1988; Smith, 1990). (McClure, 1992)

Basically, IPPs are plans outlining the personal/individual objectives a student is to achieve along with evaluation standards and timelines. The term IPP will be used throughout this study, however, it could be used interchangeably with other similar terms which are often used in the literature:

- Individual Educational Plans (IEPs)
- Individual Instructional Plans (IIPs)

Although IPPs could be used for all students in individualized programs, currently the thrust in Canada and the United States is primarily for special needs students. The reason for this can be explained by looking at the history of IPPs. IPPs, and their accompanying literature, began to emerge in 1975 in response to passing of *The Education for all Handicapped Children Act* (Public Law 94-142) in the United States (Bureau of Education for the Handicapped, 1979). This act mandated the existence of IPPs for all students in the U.S. receiving special education services. Because the enactment of this federal legislation was prompted out of public concern regarding the effectiveness of special education services (Schultz & Turnbull, 1984), the prescribed use of an accountability tool (IPPs) was included in the act.

Even though no federal statute similar to Public Law 94-142 exists in Canada, most provincial ministries of education have followed this lead with their own IPP requirements in the late 70's and 80's (McClure, 1992). In Alberta the requirements for the development of student IPPs are driven by Alberta Education's Program Policy Manual (1989). The Program Policy Manual states that school boards are responsible for the development and implementation of Individual Program Plans for all *exceptional* students. Exceptional students are defined as those who require a different program or an adaptation/modification to the regular school program. Section 29(1) of the School Act provides the guidelines a school board use to decide if an exceptional/special program is required:

> A board may determine that a student is, by virtue of the student's behavioral, communications, intellectual, learning or physical characteristics, a student in need of a special education program. (Alberta School Act, Section 29(1))

To assist School Boards and teachers with the preparation of IPPs in the province, Alberta Education Response Center (ERC) has produced the handbook *Individualized Program Plans, a Reference for Teachers* (1989). This handbook contains a guide, video tape and some sample (paper based) IPP formats.

The use of computerized IPPs grew from the emergence of micro-computer use in schools in the early 80's and the need to automate this time consuming "unrelished" process — basically educators do not cherish the production of IPPs (Ryan & Rucker, 1991). Research suggests that many educators appear to view the IPP process as something that *has to be done* to meet external requirements rather than something that *should be done* to better meet the individual educational needs of their students (Dudley-Marling, 1985). With this in mind, it is no wonder that they are looking for an automated tool to expedite the process. However the movement towards computerized IPPs has been hindered by the fact that there has been very little research done to date on the quality and effectiveness of computerized IPPs (McClure, 1992; Smith, 1990). Yet, the

meager amount that has been done does suggests that they do help to conserve time and money resources as well as have a positive influence on attitudes and "quality"[3].

One of the earliest studies on the time-saving aspect of computerized IPPs was carried out on the CAMEO (Computer Assisted management of Educational Objectives) main-frame system. Brown (1982) reported that 89% of the teachers surveyed showed reduced IPP development time when using the automated system — 50 minutes compared to the two hours when manually writing IPPs. Later studies by Ryan and Rucker (1986) and Jenkins (1987), both substantiated a reduced development time for computerized versus non-computerized IPPs[4]. However, it should be pointed out that none of these studies included the initial or ongoing time investment made by the teaching staff to learn the systems nor did they include the time taken by staff to maintain the objective databases or other parts of the system.

Studies of cost-savings through the use of computerized IPPs have been done by Enell and Barrick (1983), Enell (1984) and Ryan and Rucker (1986). Although exact costs are always difficult to determine, these studies showed a modest long-term cost reduction even when initial and ongoing software and technical assistance costs were factored in.

Studies of improvements of quality through the use of computerized IPPs are more difficult to assess — probably because *quality* is such a subjective term. Enell and Barrick (1983) reported that parents reacted more favorably to computerized plans and administrators perceived that both parents and teachers reacted more positively to the computerized plans[5]. Ryan and Rucker (1986) found that teachers who produced computerized IPPs had more favorable attitudes towards the value of the plan than did teachers who produced the plans manually. However, it should be noted that all these studies were directed towards perceptions rather than some more direct and objective measure of quality. Furthermore, how much of this favorable reaction was due to a *better* plan as opposed to the common misconception that *if it's on computer it must be better* is unknown.

The Need for a Systemic/System Approach

To ensure the successful implementation of technology for automated individualized instruction, one must use a systemic/system approach. This approach is essentially the Gestalt concept that the whole is more than a simple summation of its constituent parts. The terms *systemic* or *system approach* can be considered synonymous; they are frequently used interchangeably in the literature. This approach, when applied to

---

[3]Quality is difficult to objectively assess. This issue is discussed more fully later in the section.

[4]In this study, user of the developed system also felt that once they learned how to use the system it did save them time as well — refer to the **Post-Implementation Views and Perceptions** chapter for more details.

[5]The positive reaction of parents was also mentioned by some staff interviewed by this study — refer to the **Post-Implementation Views and Perceptions** chapter for more details.

technology, involves viewing the application of technology in a more holistic as opposed to a piece-meal manner — considering the effect this application of technology will have on other parts of the system. Gentry (1991) recognized the importance of a systemic/holistic approach in his attempt to give definition to the emerging field of educational technology. He defines **technology** as:

> ...the systemic and systematic application of behavior and physical sciences concepts and other knowledge to the solution of problems (Gentry, 1991, p. 7).

The importance of a systems approach to the application of technology has been stressed repeatedly throughout the literature — being most prevalent in the more recent literature (La Follette, 1992; Reiser & Salisbury, 1991; David, 1991; Davies, 1973). As evidenced by the frequency with which it has been addressed, this issue is a current concern with a number of educational technologists.

One area that can benefit from the systems approach to the application of technology is that of individualized instruction. Before introducing a technological solution for individualized instruction you must consider what effects this application of technology will have on other parts of the system. Not to do so is courting failure, for the introduction of progressive, but isolated, technological change has long been shown to cause unbalance in other parts of an educational system — an unbalance which can lead to failure (Lange, 1967). Failure, not because the technological change wasn't good for the system, but because of the unwelcome problems it had caused in other related areas.

> Changes that have been introduced, like programmed learning and independent study, have tended to create new educational, social, status, and administrative problems, so that people have ultimately become disenchanted with them (Davies, 1973, p. 16).

Concluding Remarks

There has been a growing concern in North American school systems and industry as to how our students and employees measure up with those in the rest of the world. There has also been a growing need for life long learning to meet the changing employment requirements in our fast paced, rapidly changing technologically-based society. Both these factors, in combination with the dramatic achievement gains possible with individualized instruction (Bloom 1984), will result in an increased movement towards this form of instruction in the future (Gentry & Csete 1991). Alberta Education (1987), in their *Visions 2000: A Vision of Educational Technology in Alberta by the Year 2000*, projected that the teacher's role in individualized instruction will increase from 5% in 1990 to 50% by the year 2000. Due to the high cost of delivery of this form of instruction through the use of "live" teachers, more pressure will be placed on technology to help solve the growing need (Clark & Sugrue, 1988). However, care must be taken when introducing an isolated technological change otherwise failure may result due to unwanted changes in other related parts of the system (Davies, 1973; Lange, 1967). A systemic/system approach is needed (La Follette, 1992; Reiser & Salisbury, 1991; David, 1991; Dick & Carey, 1990; Davies, 1973).

This study is an attempt to solve some of the problems created in another part of the educational system caused by the introduction of technology for individualized instruction (e.g. CAI) — specifically the problem of increased teacher workload (David, 1991; Alberta Education, 1987). This synergistic application of technology will result in a better, smoother, implementation of individualized instruction for both education and industry.

Chapter III

Methodology

Introduction

This chapter discusses the methodology used in the study. The specific instruments and techniques used to collect and analyze the data are discussed as well as details of the sample. This chapter also identifies and addresses all perceived ethical concerns and considerations.

The instruments and techniques used to collect the data for this study were interviews, document reviews and personal notes. The instruments and techniques used to analyze, summarize and present the resulting data were tables, figures and descriptive statistics.

Instruments and Techniques

Since this study was broken down into three research questions, the most logical way to address the specific instruments and techniques used is on a question by question basis. Details of the interview process, used to gather data for both the "System Features/Components" and "Post-Implementation Views and Perceptions" phases of the study, will also be discussed.

System Features/Components

Research Question #1: What features/components should a computerized IPP management system contain?

Two techniques were used to determine what components should be present in an automated/computerized system for the management of students in individualized programs. First, a document review of some existing systems used to manage students in individualized programs was performed. Second, a number of informal meetings and

brainstorming sessions were held with teachers and administrative support staff involved with the delivery of individualized programs. Notes from these sessions were used to generate a list of features/components considered important (refer to the guide in **Appendix C** for details of this data collection process). This list was further expanded and refined through the use of a prototyping process (see discussion of "System Development and Implementation" later in this chapter). The resulting compiled list of components was analyzed, summarized and presented through the use of tables and figures.

System Development and Implementation

Research Question #2: What is involved with the development and implementation of a computerized IPP management system?

Data for this research question came from notes kept by the researcher during the development and implementation phases of the system. These notes were then used to summarize the issues and problems encountered during this stage.

The methodology used to develop the computerized system was based on prototyping. Prototyping is a technique where the design and development phases of the traditional software development are shortened and become less distinct phases (Martin, 1984, 1989). In prototyping, initial user requirements are gathered, incorporated into a functional prototype, and then the resulting prototype is "quickly" returned to the user for evaluation and implementation (if it's a functional prototype). Then the next set of design requirements are gathered to begin the cycle again.

Prototyping has become a highly successful methodology used to develop useful computing systems. Its success is due, in part, to the "uncertainty principle" in systems development. The concept of the "uncertainty principle" was borrowed from the field of physics and quantum mechanics which states that the act of observing subatomic events changes the events being observed. Similarly, the act of providing the end users with a system they say they need changes the requirements of the system! The only way to successfully approach such a moving target is to approach it through the creation of successive prototypes — each one modified slightly to track the changing system requirements (Martin, 1984, 1989).

Post-Implementation Views and Perceptions

Research Question #3: What are the users' views and perceptions after implementation of a computerized IPP management system?

The developed IPP system (Wodelet, 1993) for managing students in individualized programs was made available to special education teachers in the TMH, EMH and ELA programs in the district (additional detail will be provided in **The Sample** section later in this chapter). Three separate half day training sessions were provided along with a detailed user manual (see **Appendix F**). Both use of the system as well as attendance at the training sessions were optional. Additional one-on-one training and consulting was available throughout the trial period either in person, by phone or through the system wide electronic mail system.

After the system had been made available for approximately nine months[6], the teachers and administrative support staff involved with use of the developed IPP system (Wodelet, 1993) were interviewed to determine their views and perceptions. Refer to the interview guide in **Appendix D** for details of this interview process. The resulting compiled list of views and perceptions was analyzed, summarized and presented through use of tables, figures and descriptive statistics.

The Interview Process

Semi-structured interviews were used because this technique provided the best balance between objectivity (through use of some structured questions) and depth (through use of probing questions). Borg & Gall (1989) and Sudman & Bradburn (1983) served as good references for developing the interview guide. These sources as well as the *University Standards for the Protection of Human Research Participants* (1991) were used to address all ethical considerations which needed to be considered when collecting these data. The identified ethical concerns are discussed more fully in "Ethical Concerns and Considerations" later in this chapter.

A pre-interview contact guide (see **Appendix B**) was developed and used prior to the main interviews (those outlined in **Appendix C** and **Appendix D**) being conducted. The pre-interview contact guide briefly covered the objective for the main interview, provided the interviewee with an estimate of the time required, re-stressed confidentiality, and setup an agreeable time for the main interview.

The researcher personally interviewed all teachers and staff for consistency and to reduce errors caused by multiple interviewers — a single-interviewer protocol helps improve both the reliability and validity of the results (Borg & Gall, 1989). Information from the interview was collected through use of a portable computer and typed directly into a pre-defined database program (the database and support sub-routines are available on the **Diskette Enclosure** included at the back of this thesis). Interviewees had the opportunity to review their summarized comments and to make necessary corrections before their comments were incorporated into the study. Interviewees were required to sign a consent form to allow use of their comments (see **Appendix A**) and were offered a summary of the results of the study, if they desired.

The Sample

There were two samples involved in this study: one of existing automated systems for the document review; the other of teachers and administrative support staff involved with the delivery of individualized programs.

---

[6]Some staff, those involved in the earliest stages of development, had the opportunity to work with the incomplete prototype for a longer period of time.

The document review was limited to a purposive sample of automated systems available and in use in the Canadian market place at the time of study. Systems available for both industry and the educational market place were considered. The initial list of systems came from advertisements in trade journals and magazines and from discussions with individuals working in the field. This list was further expanded by contacting the companies involved and obtaining a list of their competitors. Investigations indicated that the population size appears to be quite small.

The teachers and administrative support staff interviewed came from a large school district in Alberta, Canada. All staff in the district involved with the delivery of individualized programs and willing to participate in the study were interviewed. This sample contained staff from three different program areas:

Enhanced Learning Assistance (**ELA**)

Educable Mentally Handicapped (**EMH**)

Trainable Mentally Handicapped (**TMH**)

Characteristics of the student and program for each of these three groups are described below. These characteristics were derived from the "Special Needs Handbook" of the XXX School District (1992). The district is currently only implementing Individualized Program Plans (IPPs) for students in the special education area, therefore no non-special programs were involved in the study.

Enhanced Learning Assistance (ELA)

Student Characteristics

A student in Enhanced Learning Assistance exhibits an extreme discrepancy between his educational achievement and his tested intellectual potential. This extreme discrepancy may be evident in the areas of understanding or using language, spoken or written, which in turn may manifest itself in an imperfect ability to listen, think, speak, read, write, spell or do mathematical calculations. The primary causative factor in this extreme discrepancy will not include mental retardation, educational or cultural deprivation, sensory loss, or physical, medical or psychological conditions.

The major focus for the student in the Enhanced Learning Assistance program is to upgrade academic skills, teach compensatory and coping techniques and foster improved self-esteem and social skills. Students are registered in their home school in the appropriate regular classroom. The student receives special assistance in Language Arts and Math, if required. Additional assistance for other subjects may be provided through team teaching, consultation with the regular teacher, tutorial classes related to regular class subjects and modification of curriculum and student evaluation through the IPP.

Program Characteristics

The Enhanced Learning Assistance class provides an individualized program in both a small class setting and the regular classroom for the student with severe learning disabilities. Students will be registered in the regular classroom and supported by the special education teacher. Therefore, it is the joint responsibility of the regular and special education teachers to meet the needs of the student.

Individual Program Plans (IPP) will be written for all aspects of the student's program that vary from the regular curriculum. Class size is based on 10 students to 1 teacher with a 0.5 FTE (Full Time Equivalent) teacher for up to 5 students. For each additional student, a 0.1 FTE teacher is added to a maximum of 10 students. Annual reviews of placement are conducted by the school-based student support team to determine if placement continues to be appropriate and student progress is adequate.

Educable Mentally Handicapped (EMH)

Student Characteristics

The student who is educable mentally handicapped is one who is usually significantly behind others at his grade level in reading, arithmetic, and other academic subjects. Although this student generally achieves fewer academic goals than his peers, the majority, upon leaving school, should be able to hold a job, manage their affairs, and otherwise provide for themselves and their families.

The major focus of this program is to provide for the individual student's learning strengths and to develop the necessary aptitudes and skills that facilitate acceptable social performance. Students will be provided with an individualized program in the most enabling environment possible. The student can be expected to achieve academic skills ranging from the third to the seventh grade level by age 18. Students will integrate with peers in appropriate school activities and in age-appropriate classrooms as social skills permit. Most students will be able to participate with regular classes in Physical Education, Art and Music and certain units within the core areas. Students should be included in the whole spectrum of extra-curricular activities such as field trips, concerts, sports and fund-raising events with the regular class. Each student will develop the necessary skills for living and working in the normal community setting.

Program Characteristics

The primary focus of the program will be the development of social skills, numeration and living skills. Any significant lag or delay in the academic area will be taught as planned in the individual's IPP. Whenever possible, students will attend elementary school programs from the ages of 6 to 12.5 years, junior high programs from the ages of 12.5 to 14.5 years and senior high programs from 14.5 on. Student enrollment shall be small — maximum 12 students, one teacher per class and 20 hours of aide time per class.

Student progress within the program is reviewed annually and recommendations for continuance or change are directed to the Supervisor of Special Education.

Trainable Mentally Handicapped (TMH)

Student Characteristics

The student who is trainable mentally handicapped is one who is very unlikely to make meaningful achievement in the traditional academic subjects, but who is capable of profiting from instruction in living vocational skills. He may require consistent and continuous supervision of his activities for life.

The major focus of the student's program is to meet the individual developmental needs of students and allow them to achieve their maximum potential socially, emotionally and academically. Academic skills are taught according to the student's needs as outlined in the IPP. The work experience and vocational skills components of the program are increased for the older students.

Students will integrate with peers in appropriate school activities and in age-appropriate classrooms as social skills permit.

Program Characteristics

The emphasis of the program is on communication, numeration, self-care, leisure, social skills, and vocational job training. Student enrollment shall be small — maximum 8 students, one teacher per class, one aide per class. Whenever possible, students will attend elementary school programs from the ages of 5.5 to 13 years, junior high programs from the ages of 13 to 16.5, and senior high programs from the ages of 16.5 to 19.5. Annual review of student placement is conducted to determine if placement continues to be appropriate and student progress is adequate.

Ethical Concerns and Considerations

The General Faculties Council guidelines for research involving human subjects was used to identify and address the ethical concerns and considerations (University of Alberta, 1991). The specific guidelines from this document (*University Standards for the Protection of Human Research Participants*) are paraphrased and discussed below:

Guideline 1 — If research procedures could potentially produce physical or mental harm for the participant, the investigator must assess the magnitude and present justification. (Safeguarding Participants)

Since the topic of this research is relatively innocuous, no physical or mental harm to the participants was anticipated or observed. No deception was necessary in order to carry out this study.

Guideline 2 — Where possible, participants must give fully informed and voluntary consent to participation. (Safeguarding Participants)

Participation in the study was on a voluntary basis and participants had the right to opt out at any time. The nature and purpose of the study was fully explained prior to gaining their consent (refer to the consent form in **Appendix A**).

Guideline 3 — Where possible, participants must be guaranteed anonymity and their responses treated with confidentiality. (Safeguarding Participants)

Participant confidentiality was maintained throughout the study. Only the researcher had access to the interview data. Data was not kept on any file server or machine with public access. All notes were destroyed at the end of study.

Guideline 4 — Researchers must be competent in their area of inquiry, familiar with appropriate ethical guidelines, and sufficiently aware of the possible uses to which the results may be put in order to make responsible decisions. When in doubt about the application of these guidelines, the investigator is encouraged to consult with informed colleagues and supervisors. (Role of the Investigator)

From my prior experience in working with users during system development and through the creation of the research proposal and from consultation with my supervisor and other colleagues, I became sufficiently knowledgeable about the procedures and risks of the study in order to make the responsible decisions required by this guideline. All unanticipated concerns which arose during the study were dealt with through further discussions with my supervisor and informed colleagues.

Guideline 5 — The investigator must ensure that all individuals under the investigator's supervision have the training and competence needed to carry out their responsibilities. (Role of the Investigator)

The researcher was the sole investigator for this study for collection of the data from the interviews and development of the computerized system. Therefore, in this aspect, the issue of training and competence of individuals under my supervision did not apply. However, assistance was provided by some staff in the XXX School District for training, help answering users' problems, creation of the IPP disks and installation of IPP program software. Adequate training was provided for these individuals and the researcher made himself readily accessible to these assistants to deal with any unanticipated problems.

Chapter IV

System Design, Development and Implementation

Introduction

This chapter presents the results obtained from the study for the first two research questions:

1) What features/components should a computerized IPP management system contain?
2) What is involved with the development and implementation of such a system?

The results from these two research questions are difficult to separate because of the prototyping developmental approach that was used. The prototyping process, which was used for the second research question also played a major role in the generation of the compiled list of features/components for the first research question. The results from the third research question will be discussed in the following chapter, **Post-Implementation Views and Perceptions**.

Description of the Research Environment

This section provides a brief overview and description of the environment/setting in which the research was carried out. This discussion helps set the stage for the presentation of the resulting data — a proper backdrop is helpful to aid in understanding the results.

This study involved a single large school district in Alberta, Canada. The teachers and administrative support staff involved in the study came from three program areas within the district. These program areas were:

Enhanced Learning Assistance (ELA)
Educable Mentally Handicapped (EMH)
Trainable Mentally Handicapped (TMH)

A more complete description of these programs and the research environment can be found in the **Sample** section of the **Methodology** chapter.

The district is currently only implementing Individualized Program Plans (IPPs) for students in the special education area, therefore no non-special programs were involved in the study. Depending on the results of this study, IPPs may be introduced for other program areas in the future. The total number of staff and students involved in each special program is provided in **Table 1**.

Table 1: Number of Staff and Students in Program

Program	Total Staff*	Total Students*
ELA	22	172
EMH	10	93
TMH	4	34
Total	36	299

* Numbers as of January 1st, 1993

System Features/Components

This section presents the results from the first research question:

1) What features/components should a computerized IPP management system contain?

Since a prototyping developmental approach was used, the compiled list of features/components came from both a review of existing systems as well as from the prototyping developmental approach. Therefore, the results for this research question partially came from the second research question which is discussed later in this chapter in **System Development and Implementation**.

Review of Existing Systems

This review was not intended to be an exhaustive review of all available systems — that could be the topic of an entire thesis in itself. Nor was this review intended to provide a detailed description of the systems examined — each of the systems had a full manual dedicated to that purpose. Rather, this review was intended to provide some basic

information on what features/components some existing systems have for the management of IPPs. This review served as a springboard for the development of a customized system to meet the needs of the target school district.

Two existing systems for managing IPPs were reviewed. Initial investigation showed a distinct lack of computerized systems being used to manage IPPs. Had this study been expanded to review all available systems, rather than just those currently in use, a greater number could have probably been found. However, it was felt that a review of systems not currently being used, presumably because they are not meeting current needs, would not serve as good models for the development of a useful system. Furthermore, the need to examine a large number of systems is reduced because of the chosen developmental methodology. Prototyping alone would eventually produce a very effective and useful system given enough time and reiterations of the prototyping developmental loop. However, a prior review of some existing systems does help to produce a more useful and functional prototype sooner because it builds upon the initial "prototyping"[7] work done by others.

IEPWorks

This system is currently in use by the one of the largest school board in Alberta as well as a few other school districts in the province through the Board's external consulting services. It runs on the Apple IIe and Apple IIgs computers and requires AppleWorks 2.0 (Claris, 1988) or greater. The program is available from K-12 MicroMedia Publishing, 6 Arrow Road, Ramsey, NJ 07446 at a cost of $75.

Basically the system uses the word processing portion of AppleWorks along with some pre-defined keyboard driven commands to automate the copy and paste from objective files (called Goalslists) to a student's IPP. The user places the cursor on the objective to be copied and press a key. This process is repeated until all wanted objectives from the file are copied to the student's IPP. Brief on-line help is provided by pressing <closed apple>H.

Once the objectives are copied over to a student's IPP the user can use the standard AppleWorks keyboard commands to edit the text (<open apple>D to delete a block of text, <open apple>R to replace text, <open apple>1 to send cursor to the beginning of the file, etc.). The final result is a text-only document — AppleWorks does not support special graphics or advanced word-processing features such as tables. A sample IPP produced by this system is provided in **Appendix G**.

The objective files (called Goalslists) can be customized by the user. A few keyboard driven macro scripts are also provided to assist with this process. Each learning objective in the file is added in free format with the only exception being it must not contain any blank lines (blank lines are what the program uses to determine the end of an objective). Special names delimited by square brackets ("[...]") can be used in the free format text to substitute other text when the objective is copied from the data file to the student's IPP.

---

[7]Production versions of programs are not usually considered "prototypes". However, in the broad sense of the meaning and in the context of this study they are since the ideas they contain are being used to feed directly into the prototype development loop.

For example, [Fname] will be substituted with the first name of the student, [year] with the current year, [school] with the name of the school, etc.

IEP Database

IEP Database (Lunenburg, 1993) is currently in use by the Lunenburg County District School Board. It runs on Macintosh computers and requires Microsoft Works (Microsoft, 1990). The program was developed by staff within the district and, although not being actively marketed, may be obtained by contacting the Lunenburg County District School Board, P.O. Box 380, 66 Pleasant Street, Bridgewater, Nova Scotia, B4V 2W9.

The system consists of three different folders — one for *Long Term Objectives*, one for *Short Term Goals and Objectives*, and one for *Programs, Materials and Strategies*. Each folder in turn contains a number of different Microsoft Works (Microsoft, 1990) word processing files — one for each of the following domains:

- Academic
- Developmental
- Daily Living
- Personal Social
- Prevocational
- Study/Organizational

To create an IPP for a student, one opens a read-only copy of one of these files and uses the standard word processing commands to delete what is not needed. The headings Goals, Long Term Objectives and Short Term Objectives are listed only twice in a document, once at the beginning and once at the end. The user must copy and paste these headings at the beginning of each new page. Unfortunately no macro scripts are provided with this system to help automate the process. A sample IPP produced by this system is provided in **Appendix H**.

List of Features/Components

The compiled list of features/components (see **Table 2**) came from both a review of existing systems (described earlier in this chapter) as well as from the prototyping developmental approach (described later in this chapter). This list represents the culmination of many months of development and refinement. A number of features/components came from the initial systems review but at least as many, if not more, came from the prototyping process through consultation with teachers and administrative support staff. The systems review generated a loose set of ideas and direction for the project but it was the prototyping process which cemented, refined, and built upon the initial sparse framework of ideas to generate a useful and effective system.

Table 2 provides a detailed presentation of each of the identified system features/components. The goal of the discussion following this table is to review and summarize those findings and to highlight the more global themes that emerged out of the collection and analysis of these data elements. In any detailed presentation it's easy to

loose sight of the "grand picture" when standing in front of a mountain of detail. Therefore, this discussion will concentrate on the highlights and attempt to step back from the detail and discuss the results from a broader perspective.

Table 2: List of Features / Components

Feature / Component	Discussion
The management of lists of objectives is of primary importance.	This was not only the major thrust of the existing systems reviewed, it was the major concern brought out during the interviews and brainstorming sessions.
System must support / mimic the existing paper based IPP formats currently being used.	The system must give the user the ability to select between a number of standard formats currently in use (such as the check list, form report card format, etc.). The teacher should be able to switch between these different formats without loss of information. Also, if a new objective is added to the student's IPP in one format, then it should automatically be added to the IPPs in the other formats.
Users need the ability to keep different types of information (text, numbers, dates, etc.) on an objective by objective basis.	A variety of information data types needs to be kept on an objective-by-objective basis. This includes such information as expected achievement dates, test scores, free format teacher comments, etc. The system must allow entry of all possible data types such as numbers (integer and real), dates, text, money, etc. The users should be able to enter these data types in any information element for an objective and ideally be able to format the data to suit their needs.
System must be able to produce a professional looking hard copy output.	The system must be able to produce a hard copy output which can be printed on the local printers in the schools. The output must look "professional" and must be in a format which is easy for the parents to understand (unlike the output from IEPWorks). If possible, the system should also support different page orientations (landscape or portrait) to make better use of the page when long comments are entered into the data columns.

continued...

Table 2: List of Features / Components (continued)

Feature / Component	Discussion
System *must* be easy to use.	This issue came up frequently in the discussions with users of the existing systems reviewed as well as with teachers and administrative support staff in the XXX school system. Although it came up frequently, few could translate this into functional specifications — they just knew it must be easy to use! The researcher interpreted this to mean that the system must have an intuitive interface — such as one which is window-based where the user manipulates symbols and objects rather than entering keyboard commands (see **Figure 1**). Also, in easy to use systems it should be obvious to the user what can be done at any point — this can be done though the use of perpetually displayed menus or buttons. Easy to use programs should also be non-modal. This means the program's operation, behavior and menu layouts shouldn't change depending on the settings of different program modes. All options and features should be available at all times rather than forcing the user into different modes of operation requiring them to complete a mode before they can move on to something else. That way the user can execute the program options in the sequence that is the most comfortable for them. Paradoxically, this feature (making the system easy to use) was probably the hardest feature to implement in the developed system.
On-line help should be provided.	In some ways this feature is related to the "easy to use" issue. However, the degree of on-line help is inversely proportional to the ease of use — if the system is really easy to use then no on-line help would be required. The staff interviewed generally felt that if help had to be given then it should be on-line rather than requiring use of a printed manual. However, some felt a printed user manual was still important.
System should use software tools currently available in the district.	The system should be based upon, as much as possible, similar tools in use in the district. Where possible it should take advantage of the users current familiarity and knowledge of existing tools. Not only would this help reduce the development costs (which is an important issue in nearly all districts) but, more importantly, reduce the training time and give the users a greater comfort level when using the product. It is significant to note that the "greater comfort level" also ties in with the "ease of use" issue discussed previously.
System should be developed in a high level language.	Development in a high level language — preferably a 4th generation — will help reduce development time as well as help to facilitate future modifications by non-programming staff. This is also a requirement for using a prototyping developmental methodology.

continued...

Table 2: List of Features / Components (continued)

Feature / Component	Discussion
System should use a modularized, object oriented framework for development.	System should be developed using a "modularized" or object oriented framework to facilitate future changes. Since there is a high rate of change in this area, a framework that facilitates quick changes is deemed more desirable than fast execution speed.
Objectives to be added should come from a standardized database.	This would provide consistency in objectives within a school system and help reduce the problem of poorly worded and constructed objectives being presented to parents.
System should allow teachers to modify existing objectives and add new objectives.	Teachers must be able to add objectives not yet in the formal database as well as to make modifications to existing objectives (such as slight wording changes).
System should have an "add repeating" option to allow quick building of new IPPs.	Developed system should have an "add repeating" function to allow quick building of IPPs. This feature was absent in the early prototypes but as teachers became more familiar with the systems they quickly became frustrated by having to press the **Insert** button each time to add a new objective. Therefore, an "add repeating" option was added to the **Insert** dialog menu (see **Figure 4**). That way the teacher could stay in the add "loop" and could use the type-ahead feature to add a number of objectives at once.
Objectives should be separated according to program area (TMH, EMH, ELA, etc.) and allow the user to selection from any or all of the program areas.	Initially all objectives were kept in a single database for program speed and ease of use. However, during the creation of objectives for the objective file, it became apparent that the different special education groups each wanted sole control over the contents of the file. Therefore, the decision was made to separate the objectives into different files according to program area. Yet when this was done another problem surfaced — primarily for those students close to the artificial boundary between the different program areas. The problem that teachers had was that these students needed objectives from adjacent program areas. Therefore, the feature to allow teachers to easily select objectives from different program areas was added.
Give user the ability to add, change and delete objectives and to create new IPPs.	System must provide basic functions for maintaining the information on the IPP such as adding new objectives, deleting old objectives, changing existing objectives, as well as allowing the creation of brand new IPPs.

continued...

Table 2: List of Features / Components (continued)

Feature / Component	Discussion
Keyboard interface needed for the "power users".	The basic functions (changing formats, adding and deleting objectives, etc.) should be assigned to accelerator keys to allow "power users" (those who are very familiar with the system and use it frequently) to quickly access the program features without having their hand leave the keyboard. Novice or occasional users nearly always use the displayed menus or buttons to perform their actions, however more frequent users preferred to keep their hands on the keyboard as much as possible.
System should run on all types of computers.	Ideally the system should run on all available architectures such as Macintosh, Apple II, IBM PCs/clones, Digital VAX, Sun workstations, etc. This would allow teachers and support staff to make use of any available equipment at school or at home.
System should be easily customizable by the end user.	System should allow customizable page headings, column headings, variable number of columns and variable column width. Ideally, the actual operation of the program should be customizable by the user. This ties in with the use of a high level language for development. Higher level languages help to make the scripts more readable and understandable thus increasing the ability for user modification.
System should be able to work with an objective hierarchy.	Teachers and administrative support staff track different levels of objectives. Objectives vary according to level or scope — some are more global and some are more specific. As such, the objective files contain different levels of headings and subheadings for each objectives. The developed system should be able to manage sets of objectives from all these levels.
System should contain functions to allow users to easily maintain the objective files.	As the length and complexity of the objective files increased during the development of the system, so did the need for functions to automate the maintenance of the objective database. Buttons were required for renumbering, reformatting, and maintaining the different objectives levels.
Must be a way to monitor program changes and version levels.	During the development of the system it was frequently necessary to determine which installed version a user had and monitor changes between versions in order to see if a reported problem had been fixed or a requested feature had already been implemented.

Interpreting the Data

During the informal interviews and brainstorming sessions held with users to construct and refine the list of features, it was often necessary to "read between the lines" to isolate the important features. For example, during the interviews people would talk about how the ideal system "should be smart" but, upon further probing, the feature they were really trying to get at was "easy to use". They meant the system should be "smart" enough to show what can be done in a well organized, understandable way. Frequently, people could not even answer directly what features the system should contain (such as in the comment "it should be smart"). Instead, they would relate some negative experiences they had with some software packages and leave the researcher to draw meaning from this experience.

In addition, once a feature was isolated by "reading between the lines", it was often necessary to further extrapolate and translate this into a functional specification. For example, although the "ease of use" feature came up frequently, few could translate this into functional specifications — they just knew it *must be easy to use*! The researcher had to then try to translate this into a functional specification. In this case, this was interpreted to mean the system must have an intuitive window-based interface — an interface where the user manipulates symbols and objects rather than just entering keyboard commands.

It is important to stress that an effective and useful system could not have been built simply by using the "raw" results from the interviews with teachers and administrative support staff. Their comments had to be interpreted and translated by the researcher — first into a list of features and then, secondly, into a set of functional specifications.

Interrelationships of Features

As the list of features/components grew, it became apparent that these features were quite interrelated. For example:

- The "ease of use" feature is related to the "use of a high level language" feature. In addition, to implement the "easy to use" feature required the incorporation of windows and buttons. However, developing these constructs required a great deal more complex code and thus adversely effected the "easily customizable by the end user" feature.
- The "use of existing tools" feature ties in with "ease of use" feature because it builds upon user comfort level with existing tools.
- To implement the feature "easy to use" required the incorporation of windows and buttons. However, developing these constructs required a great deal more complex code and thus adversely effected the "easily customizable by the end user" feature.
- Separating the objectives according to program area adversely effected both the "ease of use" as well as "customizability by the user" feature by requiring a more complicated "Insert" dialog window (more buttons and options to choose from), more complex code (to implement the feature) and a larger and more complex user manual (to explain the feature).

Great care needed to be taken during the design and development phases of development in order to ensure that a design decision in one area did not adversely affect some feature/component in another.

Essentially, the review of existing systems and interviews produced a web of interrelated features/components. Effectively translating and massaging this web of features into a robust system by making the appropriate design decisions and tradeoffs requires a skilled system developer. Furthermore, as discussed earlier, one cannot build a list of features/components simply by using the "raw" interview data from the teachers and administrative support staff. Much original and insightful work was still required to translate their comments into a list of features and then, once this was done, into a set of functional specifications.

Prototyping versus System Reviews

The compiled list of features/components (**Table 2**) came from both a review of existing systems as well as from the prototyping developmental approach. The prototyping process proved far more useful than the systems review for generating the list of features. It may be argued that this stems from the distinct lack of computerized systems currently being used to manage IPPs. One could say that if more systems had been available and reviewed, a more extensive list of features would have been generated. This would have certainly helped, but the system review did not uncover any significant features that had not already been identified during the initial stages of prototyping. In fact, in some cases, it took the prototyping process to "find" some of the features "missed" during the systems review. For example, when IEPWorks was reviewed, the keyboard driven user interface was initially "missed" as a useful feature. In fact this feature was one of the complaints users had about this system — it was hard to remember what all the function keys did. Therefore, a "better" button/menu interface was incorporated into the early prototype. However, as user comfort level grew with the prototype, they found that what they initially thought was a "better" interface turned out to be more cumbersome because of their hands having to leave the keyboard to use the mouse. Hence, the keyboard driven interface was merged with the existing buttons and menus in later prototypes.

To summarize, the systems review was helpful in refining and confirming the importance of some of the features identified during prototyping, but prototyping played a more important role than did the review in the generation of the list of features.

System Development and Implementation

This section presents the development and implementation issues which arose during the study as well as an overview of the developed system. It presents the results from the second research question:

2) What is involved with the development and implementation of a computerized IPP management system?

The development of a functional system for the management of IPPs was the major thrust of this study. The traditional way to present the results for research and development types of research is through the use of a program listing and user manual. However, instead of directly presenting them here, they have been included in **Appendix E** and **Appendix F** respectively. The program source code, user manual and related data files are also provided in machine readable form in the **Diskette Enclosure** at the back of this thesis. **Table 33** contains the documentation of diskette contents.

Both the program source code listing and user manual are large detailed documents. By not presenting them here, as is traditionally done, there is a risk of the reader underestimating the amount of extensive and intricate work which occurred in "collecting" the data for this part of the research study. While the program listing provides a precise and detailed description of the systems operation, it would leave all but the extremely seasoned developer wanting for more information. Similarly, the user manual is much too detailed to provide a more casual reader with a satisfying overview of the system. Therefore, with this in mind, it is hoped the reader would still take the time to glance at the detail to gain an appreciation of the extensive amount of work done and find that this break from tradition helps them come away with a better understanding of the developed product.

The Prototyping Process

Prototyping was the developmental methodology used to create the developed IPP management system. This methodology was chosen because it has been shown to be more effective than traditional developmental methodologies in providing users with functional, "user-friendly" systems (Martin, 1984, 1989). Prototyping is a technique where the design and development phases of the traditional software development have been shortened and become less distinct phases. In prototyping, initial user requirements are gathered, incorporated into a functional prototype, and then the resulting prototype is "quickly" returned to the user for evaluation and implementation. Then the next set of design requirements are gathered to begin the cycle again.

The strength and success of the prototyping methodology is due, in part, to the "uncertainty principle" in systems development (Martin, 1984, 1989). The concept of the "uncertainty principle" was borrowed from the field of physics and quantum mechanics which states that the act of observing subatomic events changes the events being observed. Similarly, the act of providing the end users with a system they say they need changes the requirements of the system! User often don't know what they want until they see something like it. Prototyping becomes essentially a discovery process for the user. It provides a developmental environment which facilitates user experimentation — they are free to adjust their changing needs and wants at each reiteration of the cycle. Therefore, the ideal system is really a moving target which can only be approached

through a process of evolution rather than through a single act of creation. The only way to successfully approach such a moving target is to approach it through the creation of successive prototypes — each one modified slightly to track the changing system requirements.

During the development of this system a great many iterations of the prototyping design–development loop occurred. Sometimes these would be quick changes lasting only minutes, other times the changes would take hours or days to complete and test. During prototyping, the size of the change is unimportant — in fact the prototyping process actually favors a large number of small changes. Because of the large number of changes and the great variation in size of changes made during the evolution of the system, no tracking of the actual number of iterations was made — the important information was deemed to be not the number of changes but the changes themselves.

The Development Language

The IPP system (Wodelet, 1993) was developed using the Microsoft Excel (Microsoft, 1992a) macro language. Although this is called a *macro* language, it must not be confused with the traditional macro constructs in assemblers or the C development language (Martin, 1984, 1989).

One important consideration when using a prototyping development methodology is the choice of the development language. A 4th GL (generation language) is virtually essential because of the high level constructs that allow quick design–development turn around times. Excel has many of the features of a 4th generation development language and as such was a suitable choice for use with a prototyping developmental methodology.

> ... it is user-friendly ... a nonprofessional programmer can obtain results with it ... it employs a data-base management system directly ... it makes intelligent default assumptions about what the user wants, where possible ... it is designed for on-line operation ... encourages structured code ... non-DP [Data Processing] users can learn a subset of the language in a two-day training course ... it is designed for easy debugging ... results can be obtained in an order-of-magnitude less time than with [3rd generation languages] ... (Martin, 1984 , pp. 32-34)

Other features of the language also seemed to make it an excellent choice for the development of the management system:

- It comes with a user friendly development environment (all window-based).
- It allows, through use of the Dialog Editor, to easily construct window-based user interfaces with objects such as buttons, menus, list boxes, etc.
- It has an "English-like" command structure to help make the program code more self-documenting.
- It is an interpretive language so the effects of changes can be seen immediately instead of waiting for long save-compile-run cycles as exist for compiled developmental languages.
- It has an auto record feature to speed development by allowing the developer to automatically "write" substantial sections of code quickly and easily.

- It runs on the two major micro-computer families: Macintosh and IBM PC/clones (one of the features identified in the feature/component list in **Table 2**).

Excel is a spreadsheet program. However, because of its ability to be customized through the use of its macro language, the resulting program need not look anything like the "usual" spreadsheet (see **Figure 1**).

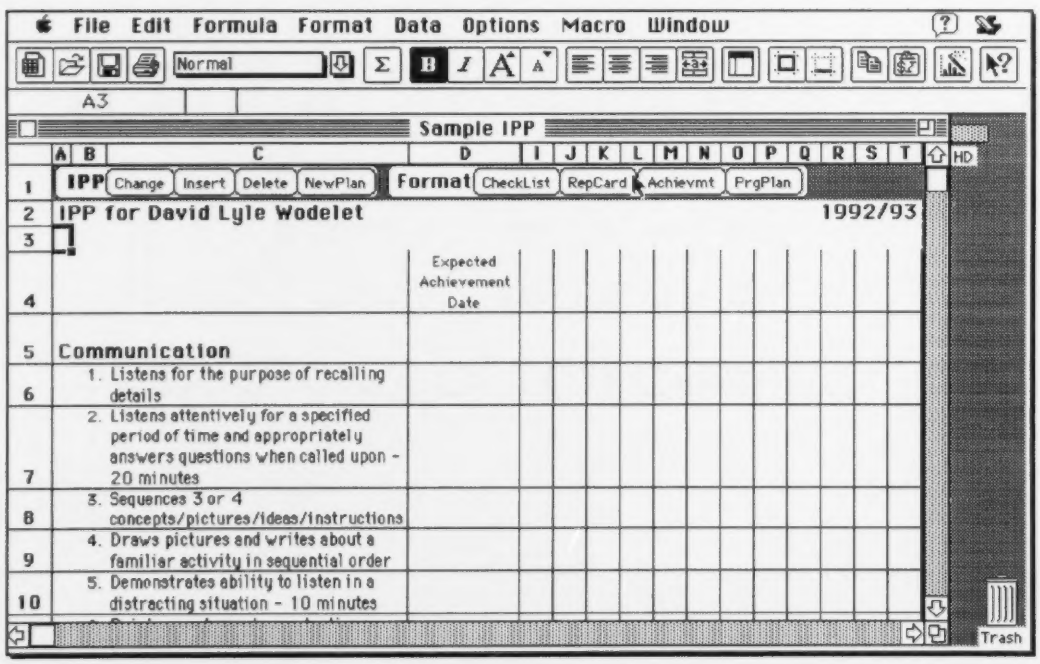
Description of the Developed System

This section provides a description of the developed IPP system. A more complete description and guide to using the system is provided in the user manual contained in **Appendix F**.

Overview

The developed system uses a window-based Graphical User Interface (GUI) — see **Figure 1**. Access to the various subroutines to manage objectives comes via the buttons at the top of the student's IPP window.

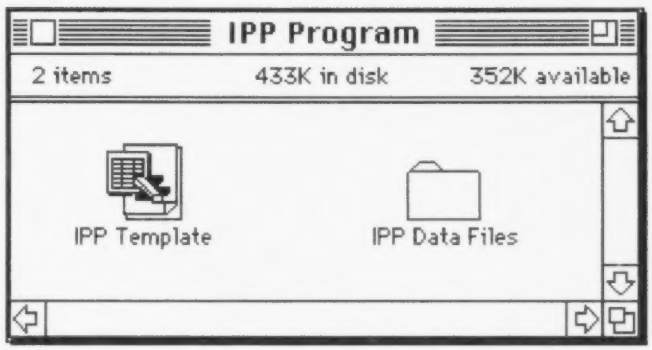
Figure 1: Sample Screen (Check List Format)


File Edit Formula Format Data Options Macro Window
Normal
A3
Sample IPP
IPP Change Insert Delete NewPlan
Format CheckList RepCard Achievmt PrgPlan
IPP for David Lyle Wodelet
1992/93
Expected Achievement Date
Communication
1. Listens for the purpose of recalling details
2. Listens attentively for a specified period of time and appropriately answers questions when called upon - 20 minutes
3. Sequences 3 or 4 concepts/pictures/ideas/instructions
4. Draws pictures and writes about a familiar activity in sequential order
5. Demonstrates ability to listen in a distracting situation - 10 minutes
HD
Trash

This IPP program will run on either a Macintosh or IBM PC/clone capable of running Windows. The system should have at least 2 megabytes of memory available but it may run with less. It may be run it from a floppy drive but a hard drive is preferred. To install the system requires least 400 kilobytes of disk space. The amount of disk space you actually need is dependent upon how may students you generate IPPs for — approximately 30 KBytes / student.

The developed system was distributed to the teaching staff through the use of 800K diskettes (see **Figure 2**) or an enclosure via electronic mail. The e-mail system proved to be a very easy way to distribute updates quickly to those having problems as well as provide help with resolving problems. However, because not all teaching staff had easy access to the e-mail system, the most common way of providing updates and help was through diskettes, phone calls and direct contact.

Figure 2: Diskette Contents



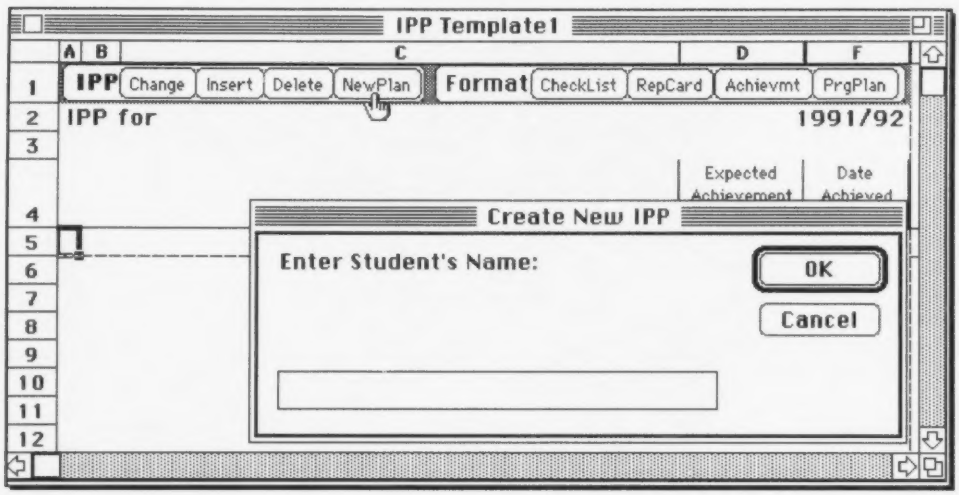
The IPP Program allow the teacher to build a student's IPP using objectives from a number of different program areas (EMH, TMH, DH, or ELA). Objectives from one or more objective lists can be combined together when creating the IPP. Once the objectives are added to the IPP the system allows the user to select from a number of different display formats. These display formats were originally patterned after some of the paper-based forms used prior to the developed system.

The computerized system developed for this research project was not intended to be a complete IPP system. Although it is fully functional and can be used as a stand-alone system, it should be used in the context of a larger framework which collects and maintains other needed information. At the XXX District it complements an existing form which collects this additional information (see **Appendix I**). This IPP information form is currently in the process of being computerized using a commercial forms package. When the IPP system is used with this form, section C is not filled out. Rather, the phrase "See Attached" is entered in this box and a printout of the student's IPP from the computerized system is attached to the form.

Creating a New IPP

There are two ways to create a new IPP for a student. The method chosen depends on whether or not the user is working on an existing IPP. If the user is already working on an IPP, then a new IPP can be created by using the **NewPlan** button on the top of the window (see **Figure 3**). If the IPP program is not already running, then a new IPP is created by double clicking on the file **IPP Template**. Either way the program will present the user with a dialog box requesting the name of the new student (see **Figure 3**). The entered name is added to the top left corner of the IPP and the current school year is automatically entered in the top right corner.

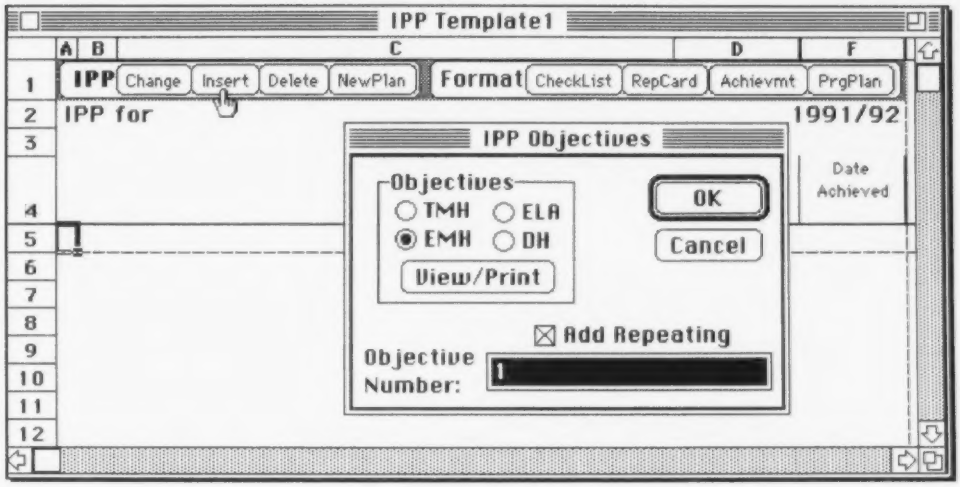
Figure 3: Creating a New IPP



Working With Objectives

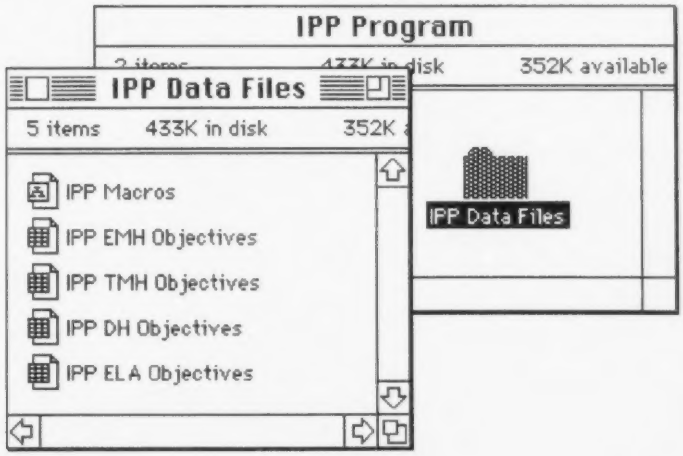
The IPP program provides users with the ability to add, delete and change objectives. Access to these functions are provided through the buttons at the top of the IPP window (see **Figure 4**). After creating a new "empty" IPP the first thing that is usually done is to add the objectives the student is expected to attain to the IPP. By pressing the **Insert** button an insert dialog box appears prompting for the objectives to be added (see **Figure 4**). The program will add/insert these objectives before the currently active row in the IPP. The currently active row is the row which contains the highlight box (the highlight box is in cell A5 in **Figure 4**). As a "user-friendly" feature the program will check to make sure that the active row is not in the titles (row 1 to 3 in **Figure 4**) or past the end of the last objective entered in the IPP. If the user was careless in the placement of the active row the program will automatically insert the objectives at the beginning (if in titles) or at the end (if past the last entered objective) of the IPP.

Figure 4: Inserting Objectives



The user can choose objectives from any or all of the four databases shown in the **IPP Objective** dialog box: **EMH**, **TMH**, **DH** and/or **ELA** (see **Figure 4**). These objective databases correspond to the separate special education programs in the XXX School District. They are actually maintained as separate database files in the **IPP Data Files** folder provided with the IPP system (see **Figure 5**). To select one of the objective databases the user simply clicks on the "circle" in front of the name of the database they want to select (**Figure 4** shows the EMH objective database selected). When this is done the program automatically removes the black highlight circle from the existing database so it is always clear to the user which database the objectives are coming from. Also, once the objective database is chosen the program automatically remembers the database "forever" so the user's preferred/default database is saved from day to day.

Figure 5: Contents of the "IPP Data Files" Folder



Viewing and Printing Objectives

The user is also given the ability to view or print any of the existing databases through the **View/Print** button (see **Figure 4**). Although it seems a waste of paper when the objective databases can be viewed on-line, most teachers seemed to find it easier to add objectives by working from a printed copy of the objective file. Pressing the **View/Print** button causes the specified objective file to be open in a read-only mode (see **Figure 6**). The read-only mode is a protection mechanism to help prevent accidental changes to the database file. This does not mean that the user cannot change the objectives in the database — for they can — but the program will not allow these changes to overwrite the original database. If the user makes changes to the database they must save the modified file elsewhere on their computer's disk. If they wish to change the original database they have to open up the actual database file directly (double click on the database file name in the **IPP Data Files** folder (see **Figure 5**)) — if it is opened it will not be made read-only. Because at the XXX School District the objective files are generated and modified by committees in each of the program areas, the group consensus was that these files should not normally be changed by the end-user. Therefore, this extra level of protection was added later on in the prototype process.

Figure 6: Sample Database File Window (EMH)

IPP EMH Objectives [Read-Only]

	A	B	C	D
1				Indent / Add / Delete / Reformat
2	1			(Type over this to insert your own objective - use the change button)
3	2			**Communication - Spelling**
4	3			**Readiness:**
5	4			Names and prints the letters of the alphabet correctly
6	5			Recognizes and writes the sounds of letters
7	6			Spells words from content areas
8	7			**Phonetic:**
9	8			Spells phonetically patterned words correctly
10	9			initial consonants
11	10			short vowels
12	11			Spells using structural word parts

Add Repeating

The **Add Repeating** check box (see **Figure 4**) was another feature which came up later in the prototyping process. By checking this box with the mouse pointer the program will repeatedly come back to this same dialog box to allow the user to add a number of objectives at a time. Otherwise, the program returns the user directly to the student's IPP window. This feature was absent in the early prototypes but as teachers became more

familiar with the systems they quickly became frustrated at having to press the **Insert** button each time to add a new objective. Therefore, the **Add Repeating** option was added to the **Insert** dialog menu (see **Figure 4**). That way the teacher could stay in the add "loop" and could use the type-ahead buffer[8] to add a number of objectives at once without continually having to wait for the window displays to catch up. Once the **Add Repeating** box is set the program automatically remembers the setting "forever" so the user's preferred/default value is saved from day to day.

Inserting/Adding Objectives

The **Objective Number** edit box (see **Figure 4**) is the place where the user types in the number of the objective they wish to add to the student's IPP. The objective number comes from the left hand side (column A) of the on-line database display window (see **Figure 6**) or from the associated hard copy output printed from this window. In each objective database objective number "**1**" is special. Entering this number causes the objective

(Type over this to insert your own objective)

to be entered into the student's IPP. This provides a convenient way for the user to enter their own unique objectives into the IPP. If minor (or even major) edit changes need to be made to existing objectives in the database then the user can make these changes directly on the student's IPP once the objective has been added. The user adds objectives by number to a student's IPP but once objectives have been added they are no longer linked by this number to the original database. This was a design decision made to allow the objective databases to be changed and reordered in future revisions without effecting any existing IPP created with older objective databases.

As objectives are added to a student's IPP they are automatically numbered to allow easy referral when discussing a student's progress with parents or support staff. A sample of this automatic numbering is provided in **Figure 1**. As objectives are added to (or deleted from when using the **Delete** button) the student's IPP, the numbers on the student's IPP are also automatically renumbered accordingly.

Changing and Deleting Objectives

The **Change** and **Delete** buttons along the top of the IPP window (see **Figure 4**) are used to change and delete existing objectives from the student's IPP. **Change** will invoke the same dialog box that **Insert** does with the exception that instead of inserting objectives above the active row (the row with the currently highlighted cell) it will replace the

---

[8]The type-ahead buffer is a standard buffer incorporated into nearly all operating systems. This buffer collects all keyboard presses made by the user (on window based operating systems it may collect mouse movements and mouse clicks as well). This buffer allows the user to type "ahead" without having to wait for the computer to finish its current task (such as a long calculation or a slow window re-display when lots of colors or intricate graphics are used). When the computer is ready for more input it simply reads the next key press (or mouse actions) from the type-ahead buffer.

existing objective. If no objective currently exists in the active row, then **Change** will behave much like an **Insert**.

As a "user-friendly" feature when using **Change** or **Delete** the program will check to make sure that the active row is not in the titles (row 1 to 3 in **Figure 4**) or past the end of the last objective entered in the IPP. If the active cell happens to be in the titles then a suitable error message is given and the change/delete is aborted. Similarly the change/delete is aborted if no objective exists on the active row — as is the case when the active row is past the end of the last objective entered.

The Objective Databases

The objective database actually contains four distinct levels of objectives. In **Figure 6**, the first two levels occur in column B and C. The third and fourth levels are contained in column D. The "objectives" in the first two levels could really more accurately be described as main headings and sub-headings respectively. The need for an objective hierarchy was identified through the prototyping process and is discussed with the list of features in **Table 2**.

The **Indent** buttons along the top of the objective database window are used to manage the fourth level of objectives since both objective levels three and four are contained in the same column. The **Add** and **Delete** indent buttons add and delete the indent used for objective level four. The **Reformat** button is used to reformat both existing as well as new objectives added to the database. Reformatting will renumber all objectives in the database and changes the font and style (e.g. bold, italics, etc.) of the different objectives levels to make the on-line and printed outputs more pleasing. The same font and style are also recreated in the student's IPP when the objectives are moved from the database to the IPP. It helps give a more professional look to the document.

Selecting an IPP Format

The format buttons along the top of the IPP window (see **Figure 7**) allow the user to select from among four different display formats. The display formats were originally patterned after some of the manual paper-based forms used prior to the developed system. Through use of the prototyping process they have evolved to their present form. There are currently four different formats the user can choose from:

Check List	(**CheckList** button; see **Figure 8** for sample)
Report Card	(**RepCard** button; see **Figure 9** for sample)
Achievement	(**Achievmt** button; see **Figure 10** for sample)
Program Plan	(**PrgPlan** button; see **Figure 11** for sample)

When the user switches between formats, none of the information entered in the other formats will be lost — it is merely hidden from view. Also as new objectives are added or deleted from the student's IPP the data elements in the format displays are also added or deleted accordingly. Therefore, the information associated with an objective in all the different formats is automatically maintained with the objective. The data elements kept in each format are not fixed. The user is free to customize each format (change headings,

add new data elements, etc.) as necessary to meet their specific needs — this aspect will be discussed in more detail later in this section in **Customizing the IPP**.

When one of the four format buttons is pressed the user is presented with a dialog box asking how the IPP will be printed (see **Figure 7**) — **Portrait** or **Landscape** (i.e. vertically or horizontally oriented on the page). Pressing one of these buttons determines how the IPP will be formatted to fit the printed page. The width of the objective column (column C in **Figure 7**) is varied in size to accommodate the full width of the printed page. When the width of this column is changed the objective it contains is also automatically re-wrapped and the row height of each column is adjusted accordingly.

Figure 7: Changing the IPP Format

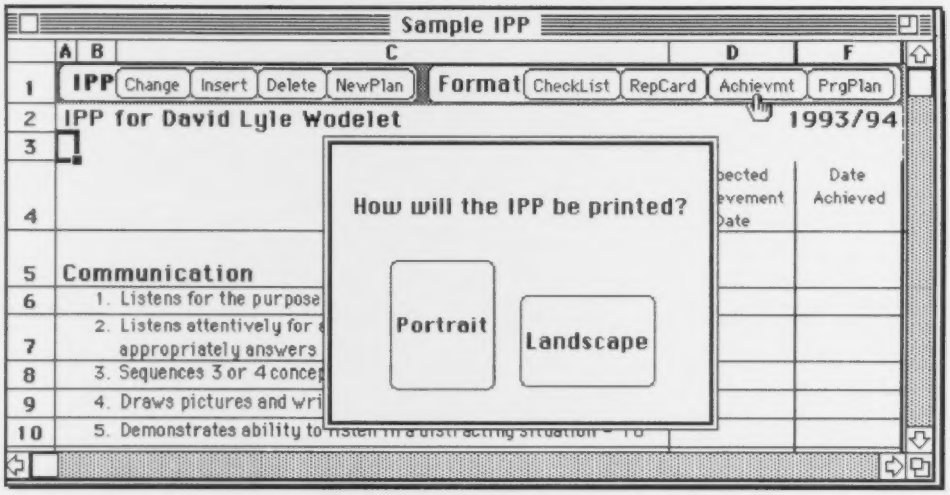


Figure 8: Sample Check List Format

IPP for <student name goes here>														1992/93
	Expected Achievement Date													
Communication														
1. Listens for the purpose of recalling details														
2. Listens attentively for a specified period of time and appropriately answers questions when called upon - 20 minutes														
3. Sequences 3 or 4 concepts/pictures/ideas/instructions														
4. Draws pictures and writes about a familiar activity in sequential order														
5. Demonstrates ability to listen in a distracting situation - 10 minutes														
Syntax/Language Concepts - Comprehends and Uses:														
1. Pronouns - me, mine, my, I you, he, she, they, his, her, their, myself, her, him, herself, himself														
2. Copula verbs - is, are, am														
3. Prepositions - up, down, top of, bottom of, in, out, on, under, above, below, beside, behind, between, in front, by														
4. Conjunctions (and, but, because, or)														
Computation														
1. When shown two sets (objects and number symbols), indicates which set contains "more" and which set contains "less"														
2. When shown two numerals indicates which is "more" and which is "less"														
3. Demonstrates with concrete materials that addition is a joining proces, using no more than 5 objects														
Science														
Matter														
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide														
Senses														
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide														

Figure 9: Sample Report Card Format

IPP for <student name goes here>				1992/93
	Expected Achievement Date	Term 1	Term 2	Term 3
Communication				
1. Listens for the purpose of recalling details				
2. Listens attentively for a specified period of time and appropriately answers questions when called upon - 20 minutes				
3. Sequences 3 or 4 concepts/pictures/ideas/instructions				
4. Draws pictures and writes about a familiar activity in sequential order				
5. Demonstrates ability to listen in a distracting situation - 10 minutes				
Syntax/Language Concepts - Comprehends and Uses:				
1. Pronouns - me, mine, my, I you, he, she, they, his, her, their, myself, her, him, herself, himself				
2. Copula verbs - is, are, am				
3. Prepositions - up, down, top of, bottom of, in, out, on, under, above, below, beside, behind, between, in front, by				
4. Conjunctions (and, but, because, or)				
Computation				
1. When shown two sets (objects and number symbols), indicates which set contains "more" and which set contains "less"				
2. When shown two numerals indicates which is "more" and which is "less"				
3. Demonstrates with concrete materials that addition is a joining proces, using no more than 5 objects				
Science				
Matter				
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide				
Senses				
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide				
Phases of Matter				
1. Reviews classifying hot and cold things				
2. Identifies solids/liquids/gases				
3. Observes that matter can change from one state to another				
4. Participates and observes in class discussions and experiments				

Figure 10: Sample Achievement Format

IPP for <student name goes here>	Expected Achievement Date	Date Achieved 1992/93
Communication		
1. Listens for the purpose of recalling details		
2. Listens attentively for a specified period of time and appropriately answers questions when called upon - 20 minutes		
3. Sequences 3 or 4 concepts/pictures/ideas/instructions		
4. Draws pictures and writes about a familiar activity in sequential order		
5. Demonstrates ability to listen in a distracting situation - 10 minutes		
Syntax/Language Concepts - Comprehends and Uses:		
1. Pronouns - me, mine, my, I you, he, she, they, his, her, their, myself, her, him, herself, himself		
2. Copula verbs - is, are, am		
3. Prepositions - up, down, top of, bottom of, in, out, on, under, above, below, beside, behind, between, in front, by		
4. Conjunctions (and, but, because, or)		
Computation		
1. When shown two sets (objects and number symbols), indicates which set contains "more" and which set contains "less"		
2. When shown two numerals indicates which is "more" and which is "less"		
3. Demonstrates with concrete materials that addition is a joining proces, using no more than 5 objects		
Science		
Matter		
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide		
Senses		
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide		
Phases of Matter		
1. Reviews classifying hot and cold things		
2. Identifies solids/liquids/gases		
3. Observes that matter can change from one state to another		
4. Participates and observes in class discussions and experiments		
Air		
1. Understands that air moves		
2. Understands that moving air can "push" things		

Figure 11: Sample Program Plan Format

IPP for <student name goes here>		1992/93
	Intervention Strategies	Evaluation Criteria
Communication		
1. Listens for the purpose of recalling details		
2. Listens attentively for a specified period of time and appropriately answers questions when called upon - 20 minutes		
3. Sequences 3 or 4 concepts/pictures/ideas/instructions		
4. Draws pictures and writes about a familiar activity in sequential order		
5. Demonstrates ability to listen in a distracting situation - 10 minutes		
Syntax/Language Concepts - Comprehends and Uses:		
1. Pronouns - me, mine, my, I you, he, she, they, his, her, their, myself, her, him, herself, himself		
2. Copula verbs - is, are, am		
3. Prepositions - up, down, top of, bottom of, in, out, on, under, above, below, beside, behind, between, in front, by		
4. Conjunctions (and, but, because, or)		
Computation		
1. When shown two sets (objects and number symbols), indicates which set contains "more" and which set contains "less"		
2. When shown two numerals indicates which is "more" and which is "less"		
3. Demonstrates with concrete materials that addition is a joining proces, using no more than 5 objects		
Science		
Matter		
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide		
Senses		
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide		
Phases of Matter		
1. Reviews classifying hot and cold things		
2. Identifies solids/liquids/gases		

Customizing the IPP

One of the strongest features of the developed system is the ability of the user to customize it to meet their own specific needs. Previous sections discussed how to insert, delete and change objectives to meet the specific needs of the student. This section discusses how the information presented by the IPP can be changed/customized to meet the specific needs of the teacher or the school system.

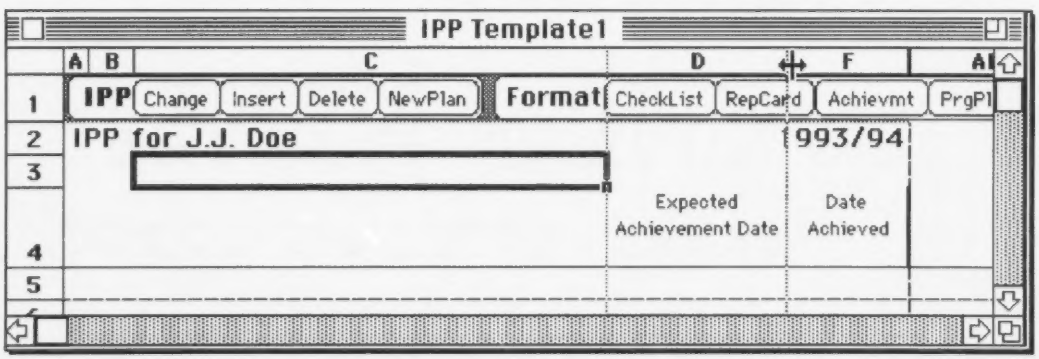
Changing Column Headings

Changing a column heading is as simple as selecting an existing column heading with the mouse and typing in the new column heading. If this needs to be done for all new IPPs created by the system, the **IPP Template**[9] file needs to be changed. The template file is a special file which the IPP program uses whenever a new IPP is created. Therefore changing the template file causes all subsequently created IPPs to incorporate the new changes. Once the template file is opened it can be modified the same way one would modify any other "normal" IPP.

Changing Column Widths

The width of any column of the IPP can also be easily changed. This is done by moving the mouse pointer into the column heading and near to the right hand side of the column, close to the column dividing line (see **Figure 12**).

Figure 12: Changing Column Widths



[9]The **IPP Template** file is a special type of file in Excel. When a template file is opened a copy of the file is automatically made and opened instead of the original file, thereby protecting the actual template from inadvertent modification. Because of this unique feature of template files, a special procedure must be used to open and modify the actual template file. The Excel manual describes template files in more detail.

The mouse pointer will change from a ✣ to a ✢ indicating the column width can be changed. Holding down the mouse button and moving the column dividing line right or left will make the column bigger or smaller. The column width of a number of columns can also be changed at once so that all of them are the same size (such as is the case in the **CheckList** format). This is done by first selecting all the columns one wants to change (click and drag over all the column headings to select them). The column with of all these selected columns can then be adjusted by changing the column width of any one of them.

After changing column width the IPP **must** be reformatted by selecting one of the format buttons at the top of the IPP window (see **Figure 12**). This will cause the IPP program to automatically re-adjust the column width of the objective column (column C in **Figure 12**) so that the IPP will again fit the full width of a single page.

The objective column (column C in **Figure 12**) is the only column width that the program readjusts automatically when changing IPP formats. When changing formats this column width is set automatically to the largest size possible depending on the width of the other columns and the page orientation. For example, when switching from a **portrait** to **landscape** page orientation, this column becomes wider to accommodate the wider page size of the landscape mode. Similarly, when the IPP is customized by changing the width of other columns in the IPP, a reformat forces the program to readjust the IPP so it again make the optimal use of the printed page.

When customizing the IPP it is important to note that there is a minimum column width enforced for the objective column. This is done to prevent it from becoming so small that the objectives in the column become unreadable. Therefore, if the other columns are made too wide the IPP may not be able to make the IPP small enough to fit within a single page width. If this happens the only solution is to reduce the width of some of the other columns or, if possible, change the page orientation from portrait to landscape.

If one wishes to change the column width of some columns for all new IPPs created then the **IPP Template** file must be changed. The template file is a special file used by the IPP system whenever new IPPs are created. It was discussed in more detail earlier in the section on **Changing Column Headings**.

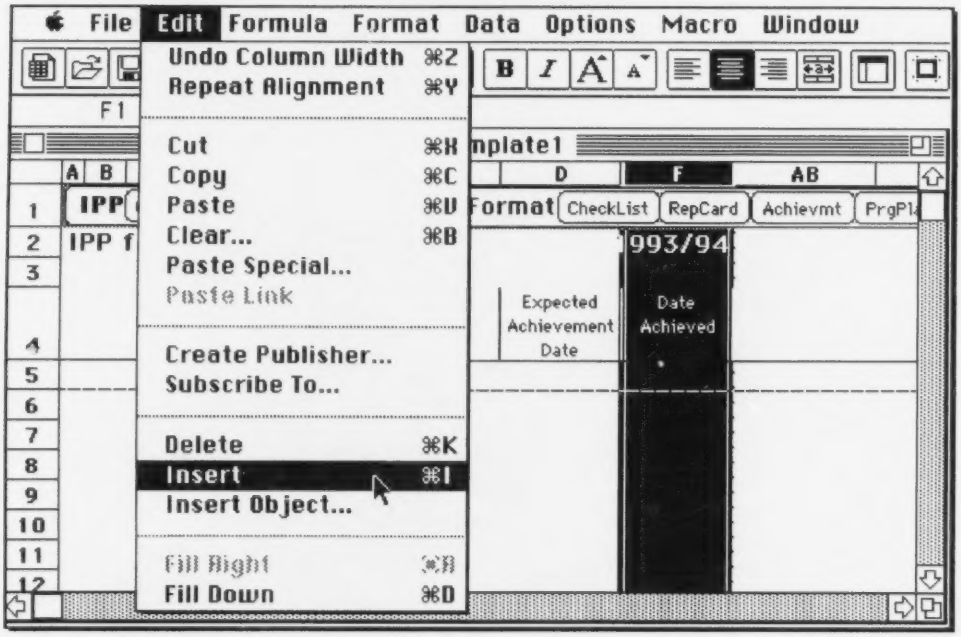
Inserting New Columns

To add a new information column to one of the existing IPP formats one needs to select an existing column which is directly **to the right of** the place where the new column should be added. Once this column is selected, the **Insert** option under the **Edit** menu is used to insert the new column (see **Figure 13**). (A short cut is to hold the option key down while you select the column to the right of where you want the new column to be — this will cause a new column to automatically be inserted.)

As was the case when changing column width, when adding new columns the IPP **must** be reformatted by selecting one of the format buttons at the top of the IPP window (see **Figure 13**). This will cause the width of the objective column to be adjusted so that the information will fit on a single page width. This was discussed in more detail previously in the section **Changing Column Widths**.

If one wishes to have some new columns added to all new IPPs created then the **IPP Template** file must be changed. The template file is a special file used by the IPP system whenever new IPP are created. It was discussed in more detail earlier in the section on **Changing Column Headings**.

Figure 13: Inserting a New Column



Changing How the Program Works

Full customizability is provided through access to Excel's macro programming language. The IPP system is completely written in this language and the complete documented source code is provided to allow it to be fully customized to meet any specific need. For example, the program source code can be modified to support additional formats and additional format buttons (as displayed along the top of the IPP window). The Excel **Function Reference Manual** that comes with the Excel program provides a complete description of the macro language. However, depending on the expertise of the user and the extensiveness of the modifications it could take anywhere from days to weeks to become fully fluent in this language. There are a number of other books and video tapes available to aid in learning the language.

The Excel macro source code is normally hidden from view to prevent unnecessary screen clutter and inadvertent modification. To access the source code file ("**IPP Macros**") you can double click on this file directly — it is in the "**IPP Data Files**" folder provided with the system (see **Figure 5**). Alternatively, if one is currently working on an

IPP, you can use the **Unhide** option under the **Window** menu to unhide the source code file. The source code is fully formatted and documented to make it easier for others to understand the program's operation (see **Figure 14**). The section **Program Layout** later in this chapter discusses the layout and organization of the program's source code.

Development and Implementation Issues

This section discusses the development and implementation issues which arose during the study. Because the development of a functional system for the management of IPPs was the major thrust of this research, it consumed a large part of the time devoted to the study. During development and implementation a great many issues arose — almost on a daily basis — which needed to be resolved. These issues, because of the use of a prototyping methodology, frequently necessitated changes or refinements to the evolving list of features/components being collected. Changes to one feature, because of the interrelationship of features, usually caused a chain reaction of changes to other features as well. The following discussion will cover some of the more significant issues and problems encountered during the development and implementation phase of the study.

Implementation of Features

Because of some unresolvable conflicts between features, not all of the features/components identified were implemented in the prototype. For example, the "easy to use" feature necessitated ruling out running on Apple II and available character cell[10] terminals attached to some of the available mini-computers because they did not have a graphical/window interface. Also, some features were not implemented due to a high cost/benefit ratio. For example, it would have taken a great deal of extra resources to develop the system to run on the available graphical UNIX workstations but this was not done due to the low numbers of staff having access to these systems. Since the majority of users had easier access to microcomputers (Macintosh, and IBM/clones) these architectures were chosen as the primary implementation platforms.

Using a High-level Language

The use of a higher level language, although virtually a necessity for prototyping, had an adverse effect on the speed of the program. It was a constant battle to keep execution

---

[10]Character cell or character-oriented terminals are "non-graphic" display devices. The firmware they contain only allows them to display individual characters at fixed row and column locations on the screen. The computer sends the terminal a binary code (usually 8 bits) of the character it wishes to display and the terminal responds by activating the pre-defined set of pixels the code represents. Any "graphic" done by the terminal must be done through the use of a special graphic character set. This set basically only allows the drawing of vertical horizontal or diagonal lines on the screen and each graphic character is bound by the same fixed character cell location as are the "normal" characters. The more advanced graphics capabilities such as pictures, curves, shading, etc. can not be done on character cell terminals.

speed at an acceptable level as more and more features were added to the system. For example, objectives were originally kept in the macro file along with the program code for fast loading and searching since this caused the fewest number of disk accesses and system page faults. However, this made it difficult maintain the objectives because of the lack of formatting functions in the macro source code file and even more difficult to print a copy of the objectives without printing the program code as well. Therefore, in subsequent prototypes, the objectives were eventually moved into their own database file and, later still, split up into many separate files according to program area. Both these changes caused a significant reduction in program startup time and execution speed. To counteract this reduction in speed, the separate objective files were loaded only on an as-needed basis rather than all at startup or when the "Insert" button was pressed. The speed could further be improved through the use of background sub-tasks — having the files loaded during the idle time when the machine is waiting for user input. However, background processing, at the time of development, was not available in the Excel (Microsoft, 1992a) macro language.

User / Developer Communication

Electronic mail, although a potentially valuable tool for prototyping, was not extensively used by users to communicate with the developer. Since the e-mail system was available during the running of the IPP system, it would have been easy for the user to simply mail comments or suggestions as they occurred and then return directly to where they left off. However, most of the users distinctly preferred a more personal contact either through telephone calls or in-person meetings. This may have been due to a lack of training on the use of the e-mail system (it was not covered during the training sessions).

Program Updates

During the prototyping process, updates are frequently sent out to users in order to fix existing problems (bugs) and provide additional functionality as new features are implemented. Therefore it is essential that the program and data files are kept in a form that allows easy user installed upgrades (where possible) to the new versions of the software. This was done by providing the user with a single diskette containing the new version. The diskette contained two files — the template file for creating new IPPs and a file folder containing the program and the associated database files. The database files (containing the objectives) were themselves contained in another folder within the program folder. To update to the new version the user simply had to copy the entire contents of the diskette to their local folder containing the students IPPs. The old version of the template file and program folder would simply be replaced. This distribution format provided the greatest ease of use and flexibility when distributing new versions since all program related files could be updated at once and none of the users existing student IPPs would be effected. The decision to keep all the database files in a single folder was made to allow updates to just the database alone. However this was never done during the study since it was just as easy to provide a complete update.

Backward Compatibility

An important consideration during the prototyping process is to provide backward compatibility as much as possible between versions. That way existing student IPPs created with older versions will still continue to work properly. However, this was not possible in all cases. For example, the decision to support a hierarchical objective structure required a number of fundamental changes to the IPP format. There was no way to make the old format compatible with the new changes. Therefore, special code had to be added to the new prototype to automatically reformat all old student IPPs to the new format. This was setup to run automatically the first time an older IPP file was opened by the new prototype.

Program Version Management

Because of the frequency of updates during the prototyping process, it was important to be able to determine which installed version a user had in order to see if an existing problem had been subsequently fixed. To do this, a version numbering scheme was used. It consisted of two parts, a major version number and a minor version number. The minor version number was incremented each time a change was made to the system. The major version number was only changed for major changes in program functionality. When the major version number changed the minor version number was reset to zero. During this study, the major version number started out as zero for pre-production (not fully functional) versions of the prototype and changed to "1" when the system was made available for general use.

Also a log of changes, known bugs and bug fixes was maintained by the researcher to assist when discussing problems with users.

Use of Templates

The IPP system uses a template mechanism to create new student IPPs. The other alternative would have been to add all the commands necessary to build a new IPP to the "Insert" button. This would have the advantage of being faster and require the use of one less data file (the template file) within the program folder. However, it would have the distinct disadvantage of being much more difficult to modify since it would require the user have a working knowledge of the macro language and, more importantly, changes would not be immediately observable. For example, to simply bold a cell or change the column width would have required the user to first find the code in the macro file which formatted that part of the IPP, then determine the proper command necessary to make this change. Once this was done, to see if the proper change had been made would have involved running the program to set up a new IPP. The use of a template file requires considerably less knowledge of Excel (Microsoft, 1992a) to make the appropriate changes — the novice can simply use the pull-down menus to make nearly all possible format changes — and the changes become immediately apparent. The use of a template file seemed to be an good design decision in order to honor both the "easy to use" and "user customizable" features.

Program Automation

Much effort was made during the development of the system to automate as many features as possible. For example, an "auto-open" feature was used to automatically open the macro subroutines whenever an existing student's IPP was opened. In addition, every time a student's IPP was opened, a check was made to see if it contained any student data. If not, as is the case when the template file was opened to create a brand new IPP, then the subroutine to create a new IPP was automatically called to prompt for student's name and other required information. Other information in the student's IPP, such as the date and school year were automatically initialized without prompting. Also, existing student IPPs were automatically upgraded from old to new prototype formats whenever old IPPs were opened (this aspect is discussed more fully in the section **Backward Compatibility**).

Safeguarding the System

To reduce screen clutter and to prevent inadvertent modification to the program code, the macro source file window is automatically hidden from view. A special subroutine in the source code file is called every time the program is opened (it's automatically gets opened whenever the teacher accesses a student's IPP). One of its functions is to ensure the program source code window is, by default, hidden from view. This, unfortunately, ran counter to the feature to make the system "easily customizable by the end user" because it makes it more difficult to find the source code, but it supports the "ease of use" feature because it makes the system seem less complicated as there are fewer windows open to confuse the user. Hiding the macro source code seemed an appropriate tradeoff since those modifying the source code would have to have a more extensive knowledge of the system so would probably be able to figure out how to "unhide" the window — the "Unhide Window" option is in one of Excel's (Microsoft, 1992a) standard pull-down menus.

Safeguarding the Student's Data

The confidentiality of student's IPP data is always an important issue for educators. Most teachers kept their students' IPP data on a single diskette which they took with them. Therefore, no data were kept on machines with public access. For those teachers who had their own computers, or those who chose to keep their student files on shared machines, document password protection was available. Document password protection is a standard file save option built into the Excel (Microsoft, 1992a) program so no specialized code needed to be added to the developed system in order to implement this feature.

Training and User Support

During implementation, *hands-on* training as well as on-going user support were two very important considerations. The training sessions were well attended and most users

were satisfied with the material covered (discussed more fully in **Post-Implementation Views and Perceptions**). The on-going support was well subscribed to and, from user comments, seemed more important than the actual group training to the success of the project. This is understandable in the sense that most people don't start to push the limits of the system and their understanding until confronted with some unique requirement of their students or school.

Modular System Design

The use of a modular system design — through the use of subroutines — proved to be a very valuable design approach during the development of the system:

- It reduced the size of the program source code through reuse of common functions thereby increasing speed (fewer disk accesses and system page faults) and improving maintainability (fewer lines of code to support).
- Modularity improved turn around time for problem (bug) fixes. It enabled the developer to track down problems more quickly since the modularity helped rule out larger blocks of code at a time.
- Modularity helped increase the robustness and reliability of the entire system. Once a problem was fixed in a subroutine, it was fixed for all other portions of code that called that subroutine. Unfortunately the converse is also true — if you break some code then all other parts of the system that call that common code also break. However, this still helped to improve reliability since you would quickly find out when something didn't work because a number of separate parts of the system would be broken at once. This increased the chance of finding a new problem during the testing phase.

Shared Variable Name Space

One serious problem encountered during development was that of a shared global name space for variables. The problem with a shared global name space is that if you use the same variable name in two different subroutines, changes to the variable contents in one subroutine also changes it for the other subroutine — which usually causes unwanted deleterious results. This is sometimes viewed as a "feature" of a programming language because it enables all subroutines to access all data used in the entire program. However, this is generally considered more of a detriment than an asset and is vehemently opposed by proponents of structured programming techniques.

To get around this language inadequacy a variable naming scheme was implemented. All variable names used within a subroutine were prefixed with a unique 3-4 letter code (usually an acronym of the subroutine name). This prefix was used to ensure unique variable names throughout the entire program. This naming scheme was further extended to include the use of references (e.g. ar<VariableName>) and temporary variables (e.g. a<VariableName>). This naming scheme made it easy to view and track all references and temporary variables during debugging since all the variables show up together in the "Define Name" dialog box provided by Excel (Microsoft, 1992a). It was easy to trace variable contents (the "a" caused them to show up first in the dialog box) and to check to make sure all memory used for variables was released at the end of a program.

Variable Creation and Destruction

In the Excel (Microsoft, 1992a) macro language variables are implicitly created the first time they are used and can contain any of the data types supported by the language (text, integers, floating point numbers, references, logicals, etc.). This is an excellent feature and assists with making the program "easily customizable by the end user" (one of the features from the list of features identified). However, all variables (whether implicitly created or not) must be explicitly released at the end of a subroutine in order to make the memory they used available to the rest of the system. This "feature" is not so "user friendly" and requires extra care by the user when modifying the program — or lots of memory!

Subroutine Arguments

The Excel (Microsoft, 1992a) macro language allows the user to create subroutines that can accept arguments and return results. This helps encourage program modularity (see section **Modular System Design**) and improves maintainability since it makes it obvious what data the subroutine requires — it's listed in the arguments of the subroutine call. However, most of the arguments for Excel's (Microsoft, 1992a) built-in functions have been implemented as numbers rather than through the use of text mnemonics. For example, to find out if the current document is read-only requires the subroutine call GET.DOCUMENT(5). It would be better if some mnemonic was used instead such as in GET.DOCUMENT(Read_Only). This would make the argument lists for function calls easier to construct and make it more self-documenting.

There is a similar problem with the use of numbers rather than mnemonic names in dialog boxes. Therefore, where possible, variable names were used to define these constants. The naming scheme used for these "constants" was "c<VariableName>". Excel, to be more user friendly, should have pre-defined constant names (like in "C" and assembler languages) for all the constants it uses.

Program Layout

Program layout is very important when developing any system to facilitate future maintenance and modification — especially if the intent is to facilitate future modification by non-programming staff (the "customizable by the end user" feature). To facilitate this a three column approach was used to lay out the macro source code (see **Figure 14**).

The first column is used to keep (and self-document) reference names which are used for the target of the GOTO function or to reference a return value from a function call which is needed elsewhere in the program. Even though the names are contained in the first column they actually reference the adjacent cell in the second column. The second column contains the actual macro source code. The comments and documentation for each line of source code is contained in the third column. Because Excel (Microsoft, 1992a) makes heavy use of numbers rather than names in function calls (discussed in the section **Subroutine Arguments**) the documentation is essential for understanding the flow of the program code.

Figure 14: Program Source Code Layout

names	*commands*	*comments*
	auto_open	
arThisMacroFileN	=GET.CELL(32,A4) =FULL(TRUE) =WINDOW.SIZE(560,420,arThisMacroFileName) =RETURN()	*get name of this document* *reverse size change by auto_clos*
	IPPNewPlan	*Creates a NEW IPP Plan/worksh*
arNewStudentNam	=ECHO(FALSE) =INPUT("Enter Student's Name:",2,"Create New IP =IF(arNewStudentName,,HALT()) =CreateNewPlan(TRUE) =RETURN()	*no screen update for speed* *get student's name* *quit if Cancel button pressed* *Create the new plan*

User Input Layout

The dialog box is the main mechanism used by Excel's (Microsoft, 1992a) macro language to get input from the user. This interface allows the developer to access all of the standard graphical-based objects in a windowing environment — buttons, pull-down menus, check boxes, edit boxes, radio/option buttons, etc. Because of this, the dialog box is probably one of the most complex functions provided. To make it easier to use and, more importantly, to facilitate customization by others, a well documented layout is essential (see **Figure 15**).

Figure 15: Dialog Box Data Table Layout

DIALOGS							
type	x	y	wide	high	text	init/result	*names*
udGetIPPDia							
			250	129	IPP Objectives		
1	178	12	64		OK		
2	178	43	64		Cancel		
7	4	101	171			1	*ud.IPPcode*
5	2	83			IPP Code:		
13	84	81			Add Repeating	TRUE	*ud.AddRepeating*
14	12	6	118	61	Objectives		
11						2	*ud.TMHorEMH*
12					TMH		
12					EMH		
12	79	24			ELA		
12					DH		

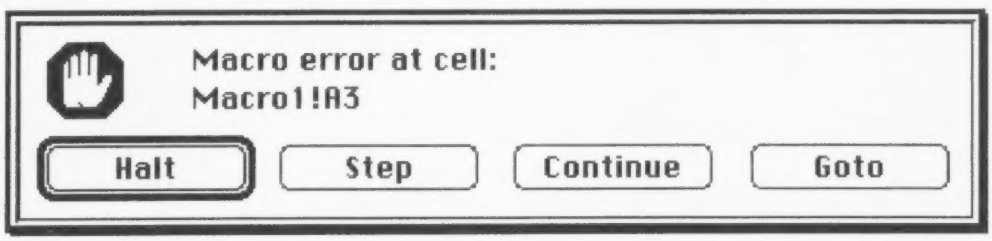
The top left corner of the dialog box data table contains the defined reference name which is used by the DIALOG.BOX() call in the actual source code — this helps make the call more self-documenting. Headings along the top of the data table document the contents of each column in the data table. The name column contains reference variables which refer to the contents of the "init/results" column of the dialog data table. The "init/results" column is where the DIALOG.BOX() function accepts initial values for each field in the dialog and returns the information entered by the user. The use of variables to refer to these data helps to make the program source code self-documenting and easier to follow.

The information in the dialog data table can be updated directly to change the types of objects (buttons, pull-down menus, etc.) displayed, where they are displayed, and to control the objects operation. However, because of the extensive use of numbers rather than mnemonics to control the dialog box, this is a very cumbersome process. A separate program — the Dialog Editor — can be used to facilitate end user modification. The clipboard is used to copy and paste the dialog data table between the macro source code file and the Dialog Editor. The Dialog Editor then provides the user with a window-based graphical user interface to create or update all dialog objects — it hides the "raw" numbers, pixel offset details, etc. from the user. This is certainly the best way for novice users, and even seasoned developers, to construct the data tables for dialog boxes.

Source Code Debugging

Development of the prototype and the fixing program problems which arose during testing was severely hampered without the support of an integrated source code debugger. When an error was encountered during the running of the program it would simply abort with a terse and unhelpful message (see **Figure 16**). It was left up to the developer to decide if the arguments were wrong, some variable contained invalid data, etc. The message was certainly not helpful enough to enable novice users to track down and fix problems — especially when they are more likely to have problems which need to be corrected. The lack of adequate debugging support is a serious drawback in making a program "easily customizable by the end user" (one of the features identified in the list of feature).

Figure 16: Macro Error Message



Chapter V

Post-Implementation Views and Perceptions

Introduction

This chapter presents the results from the third research question:

3) What are the users' views and perceptions after implementation of a computerized IPP management system?

The results from the first two research question were presented in the preceding chapter, **System Design, Development and Implementation**.

After the developed IPP system (Wodelet, 1993) had been made available for approximately nine months[11], the special education teachers who had been trained on its use were interviewed to determine their views and perceptions. The interview guide in **Appendix D** was used to conduct this interview process. Semi-structured interviews were used because this technique provided the best balance between objectivity (through use of some structured questions) and depth (through use of probing questions) (Borg & Gall, 1989; Sudman & Bradburn, 1983).

Interview Summary

The number and percentage of staff interviewed in each program area is presented in **Table 3**.

---

[11]Some staff, those involved in the earliest stages of development, had the opportunity to work with the incomplete prototype for a slightly longer period of time.

Table 3: Staff Interviewed for Post-Implementation Views and Perceptions

Program	Total Staff	Total Interviewed	Percentage Interviewed
ELA	22	14	64%
EMH	10	7	70%
TMH	4	3	75%
Total	36	24	67%

In the rest of the tables contained in this chapter the reader may notice that the number of respondents for a question may not be the same as the total number interviewed. This is because not all staff were required to answer all questions. Those staff which did not use the developed system only answered a subset of the total questions. For example, it did not make sense to ask respondents who did not use the system how frequently they used the system or how they rated the quality of training sessions they did not attend. The total responses collected on a question by question basis is summarized in **Table 4**.

Table 4: Total Responses per Question

Question	Program			Total Responses	%
	ELA	EMH	TMH		
1	14	7	3	24	100
2	2	3	3	8	33
3	2	3	3	8	33
4	2	3	3	8	33
5	14	7	3	24	100
6	14	7	3	24	100
7	12	5	3	20	83
8	2	3	3	8	33
9	2	3	3	8	33
10	2	3	3	8	33
11	2	3	3	8	33
12	14	7	3	24	100
13	2	3	3	8	33
14	14	7	3	24	100
15	14	7	3	24	100
16	14	7	3	24	100
17	2	3	3	8	33
18	14	7	3	24	100
19	2	3	3	8	33
20	14	7	3	24	100

Who is Using the System

Approximately one-third of the staff interviewed (8 of 24; **Table 5**) actually used the developed system. The use of this system was not mandated by the school board, so its use was on a purely voluntary basis. Therefore, considering the extra time and work required to learn a new system, this is not a disappointing result. The most common reason given for not using the system was related to difficulty in getting access to a computer (**Table 6**). The second most common reason was lack of time (too much work or seemed too difficult / take too much time to learn) it was easier and quicker to do it by hand[12]. A look at some of the actual comments made during the interview process helps to best illustrate the views held by those who did not use the system:

> "I prefer the handwritten [IPPs]" — "I like the old fashioned one where you write and your own thoughts go into it. I cannot see the benefit of it on the computer" — "looks like a lot more work; for simplicity teachers are always looking for ways to cut down on work; don't mean to sound lazy but if it doesn't save work then why do it?" — "I'm getting [close] to retiring... If I can get along without it until then, good" — "I don't like computers at all; I'm very impatient working with computers; get quite uptight when working with them" — "I would do it by hand because that's easier and faster" — "What I'm doing now is very easy, streamlined and very efficient. I find it difficult for a computerized IPP to make my life more simpler... Everything now is in a binder or in my head and I don't think anything could be simpler than that." — "I don't think even changing it would make me want to use it. I'm very comfortable with what I'm doing now [hand writing IPPs]" — "I find computers a hassle I don't need"

Some of the above comments suggest computer literacy was a significant deterrent to use of the system. While no one explicitly said, *I'm computer illiterate and that's why I didn't use it*, a number of comments implied that they were afraid of computers or didn't have sufficient knowledge to use them easily or effectively. A computer literacy problem was also alluded to in some of the comments with respect to the training given — a need for more training on basic micro-computer skills (**Table 22**). Question 16 (see **Table 7**) tried to get at the issue of computer literacy by having respondents rate their comfort level in working with computers. It is interesting to note that more respondents (11) rated their comfort level below average (rating level of 3) than did above average (6). Of the 11 respondents who rated themselves below average, 8 did not use the system. It is also interesting to note that 7 people rated themselves as being very uncomfortable (5 - the lowest possible value) with computers. Of these 7 nearly all (6) did not use the system.

On the average, the users of the system seem to be slightly more comfortable with computers than those not using the system. The average comfort level rating for those using the system was 3.1 (average comfort) where those not using the system had an average comfort level of 3.6 (slightly uncomfortable). However the number of individuals involved in the study was not great enough for these differences to be statistically significant.

---

[12]Being "quicker and easier to do by hand" is also partly related to the inadequate computer access — if it's not easy or convenient to get access to a computer, then it certainly is easier to do it by hand. Having better computer access would certainly help to alleviate this concern.

Table 5: Question #1 — Did you use the IPP computerized management system?

Response	ELA	EMH	TMH	Total
Yes	2	3	3	8
No	12	4	0	16
Total	14	7	3	24

Table 6: Question #1 — Comment Summary of Reasons For Not Using the System

Comment Category	Count
Difficulty getting access to a computer.	6
Looked like it was too much work to use. / Looked too complicated. / Quicker and/or easier to do by hand.	5
Already had them done or started when it came out.	3
Didn't have enough time (learn it or spend on it to do it).	3
Didn't like the objectives that come with the system.	2
Not comfortable with computers and/or program.	2
Had problems with training. Didn't learn enough about system to use it.	1
Liked the old-fashioned way where your own thoughts go into it	1

Table 7: Question #16 — How would you rate your comfort level in working with computers?

System Rating	Number of Respondents				Using System?	
	ELA	EMH	TMH	Total	Yes	No
1(very comfortable)	1	1		2		2
2	3	1		4	3	1
3	4	1	2	7	2	5
4	2	2		4	2	2
5(very uncomfortable)	4	2	1	7	1	6
Total	14	7	3	24	8	16
Average Rating	3.4	3.4	3.7	3.4	3.1	3.6

How the System was Rated

Overall, those who did use the system rated it very highly (**Table 8**). Nearly everyone (7 of 8) indicated they felt the system was very useful (rating level "2") the other person felt the system was extremely useful (rating level "1"). However, this rating level cannot adequately reflect how the users actually felt about the system. A better understanding can be gained through a look at some of the comments made during the interviews:

> "100% better than what we used to do" — "easy to use" — "I find that this system meets my needs and works great" — "good to have a database" — "Parent reaction has been great" — "Parents seem to respond better to this than the old handwritten IPP" — "I like what I see" — "[There is] nothing wrong with the software — a lot of work has gone into it" — "What's there is pretty good" — "It's a lot easier to have it on computer rather than [a] hard copy" — "the way it [the system] is now isn't too bad" — "I think it's really good" — "I really like the objectives in the database so I don't have to type them in by hand" — "The beauty of the program was that I could edit as I wanted to individualize it for my purposes" — "It's such a professional copy" — "It's easy to go back and keep track of objectives" — "Output looks good; easy to read; simple to look at and find information" — "It's nice to have all the information in front of you and to pick and choose the objectives" — "[It's] convenient to use and faster than doing it by hand" — "I like it because I can modify it to meet my needs" — "looks very professional" — "I found it fairly straight forward to understand. I liked being able to make changes" — "I like it [the system] because it is so user friendly" — "I think it makes things so much easier. Less writing,

objectives are more to the point, and parents respond better to checkpoints. [It's] easier for them to visualize where the student is and where he is going." — "Compared to [the] manual method it saved me a lot of time" — "Worked very, very well for me... Can't imagine writing it all out by hand anymore" — "I thought it was very good" — "I found it more interesting to work with on the computer than doing it by hand... Got a better product in the end" — "Gives a nice professional look" — "I found parents to be very receptive to the program... This is the only computer thing I've done." — "This is the only way that I do reporting now — I think it's great" — "Really pleased with it" — "Best IPP I've worked with because it's a working [on-line] document"

Everyone who used the system felt that, once they had learned to use the system, it saved them time (**Table 9**). This is in agreement with the finding of Ryan and Rucker (1986) and Jenkins (1987) — both substantiated a reduced development time for computerized versus non-computerized IPPs. Neither one of these studies, like this one, included the initial time investment made by the teaching staff to learn the system.

Table 8: Question #11 — How would you rate the system overall?

System Rating	ELA	EMH	TMH	Total
1(extremely useful)	0	1	0	1
2	2	2	3	7
3	0	0	0	0
4	0	0	0	0
5(not useful at all)	0	0	0	0
Total	2	3	3	8

Table 9: Question #9 — Once you learned how to use the computerized IPP system did it save time?

Save Time	ELA	EMH	TMH	Total
Yes	2	3	3	8
No	0	0	0	0
Total	2	3	3	8

How the System is Being Used

It appears that most felt it was better to start using the system at the beginning of the year rather than during the school term. Nearly all teachers who used the system started at the beginning of the year (7 of 8; **Table 10**).

One surprising result was the infrequency of use the teachers made of the actual computerized system (as opposed to use of the data forms which it could produce[13]). Of those who used the computerized system, half (4 of 8; **Table 11**) had only used it "once"[14] at the beginning of the year for the initial setup. The other half used it at the beginning of the year for setup and then again every couple of months to update their student objectives. Those teachers who only used the system once at the beginning of the year, thereafter made "pencil" additions and changes to their IPPs on the hard copy output produced by the system. The infrequent use of the computerized system seemed mainly due to problems with computer access. In the case of one user, it was because he/she was so uncomfortable using the system that it was less stressful to do it by hand.

In contrast to the actual computerized system, the data sheets produced by the system were used more frequently. Most (5 of 8; **Table 12**) used these data sheets monthly or more often to monitor and update student progress. Another two teachers indicated they used them every couple of months. The reason given for this frequency of use was that they had only a small number of students and "You get to know the kids fairly well so have a good idea where they are at, so don't need to look at it that frequently". Also, the problem with computer access came up again "Getting access [to a computer] is a problem", "If I had a computer on my desk would use it more frequently".

Note that one teacher indicated that he/she had now "stopped" using the data sheets. This is because the data had been moved to the companion IPP form (see **Appendix I**) which he/she was piloting. However, due to the inferior objective management capabilities of the piloted system, this teacher is planning to move the objectives back again to the original system for the following year.

Of those teachers who were using the system, they did so for all their students (**Table 13**). The average number of students the teachers wrote IPPs for was approximately 10 (**Table 14**). The number ranged from 6 to 20 but the most frequent number was 11 (6 of the 24 respondents said they wrote IPPs for 11 students). There seemed to be no correlation between the number of students a teacher had to write IPPs for and whether or not they use the system — the average number of students teachers wrote IPP for was the same for those who used the system and those who did not (approximately 11 students **Table 14**).

---

[13]There were two questions aimed at determining frequency of system use. Question #3 (see **Appendix D**) was used to determine the frequency of use of the computerized system (i.e. how frequently they actually used the computer to work with the IPP data). Question #4 was used to determine the frequency of use of the data produced by the system (i.e. the various formatted data sheets that the system could produce).

[14]"Once" is a relative term. The "once" refers to the initial setup that occurred <u>once</u> at the beginning of the year but required a number of computer sessions over a period of a couple of weeks. None of the teachers were actually able to finish all their IPPs in a single computer session.

Although the number of students does not seem to be related to whether or not the teacher uses the system, it looks like the number of objectives may. On the average, those who used the system kept approximately 80 objectives on a student while those that did not kept approximately 10 (see **Table 15**). These results could be interpreted as meaning that teachers with a large number of objectives to manage are more likely to use the system. Alternatively, though, it may be that users keep a larger number of objectives per student because they now have a tool that makes this feasible. More research is required to determine which interpretation, if either, is correct.

Everyone who used the system said they had to change some of the objectives provided in the databases (**Table 16**). The need to change objectives had been identified during the prototyping process, therefore the system was designed to give the user total control over the changing of objectives — but **only after** they had been added to a student's IPP. The users were not given the same easy ability to change the objectives on the database[15]. It was felt by the users that the databases should remain standardized — they could still be changed but only by going though the group meeting held in each of the program areas. It appears that this was a good design decision since most of the changes made by the users to the objectives were relatively infrequent (**Table 17**) and mainly minor in nature (**Table 18**).

Table 10: Question #2 — When did you start using the IPP system?

Date	ELA	EMH	TMH	Total
Aug. 92		1		1
Sep. 92	1	2	3	6
Oct. 92	1			1
Total	2	3	3	8

[15]The objective databases are opened read-only when accessed by the user to prevent changes. However it is still possible for the knowledgeable user to change the wording of objectives in the database, but this ability was not documented in the user manual.

Table 11: Question #3 — How often do you use the actual computerized system?

Usage	ELA	EMH	TMH	Total
Once; initial setup	2	1	1	4
Every Couple Months		2	2	4
Total	2	3	3	8

Table 12: Question #4 — How often do you use the data sheets provided by the system?

Usage	ELA	EMH	TMH	Total
Monthly or More Often	1	2	2	5
Every Couple Months		1	1	2
Stopped	1			1
Total	2	3	3	8

Table 13: Question #13 — Of the students you need to write IPPs for, approximately what percentage these students did you put on the system?

% put on System	ELA	EMH	TMH	Total
25%	0	0	0	0
50%	0	0	0	0
75%	0	0	0	0
100%	2	3	3	8
Total	2	3	3	8

Table 14: Question #14 — How many students do you write IPPs for?

Number of Students	Number of Respondents				Using System?	
	ELA	EMH	TMH	Total	Yes	No
6	1	1		2		2
7	1		1	2	1	1
8	1	1	1	3	1	2
9	1		1	2	1	1
10	1			1		1
11	4	2		6	2	4
12	2	2		4	1	3
13		1		1	1	
16	1			1	1	
20	2			2		2
Total	14	7	3	24	8	16
Average Number of Students	11.7	10.4	8.0	10.9	10.9	10.9

Table 15: Question #15 — On the average, how many IPP objectives do you maintain on a student?

Number of Objectives	Number of Respondents				Using System?	
	ELA	EMH	TMH	Total	Yes	No
4	2			2		2
5	1	3		4		4
6	1			1		1
9	1			1		1
10	1	1		2		2
12	2			2		2
13	1			1		1
18	1			1		1
20	1			1		1
30	1			1		1
33		1		1	1	
50		1	1	2	2	
60		1	1	2	2	
80			1	1	1	
150	1			1	1	
170	1			1	1	
Total	14	7	3	24	8	16
Average Number of Objectives	33.1	24.0	63.3	34.2	81.6	10.5

Table 16: Question #17 — Did you need to change the objectives provided in the database?

Needed Changes	ELA	EMH	TMH	Total
Yes	2	3	3	8
No	0	0	0	0
Total	2	3	3	8

Table 17: Question #17 — Frequency of Changes to Database Objectives

Changes to Objectives in Database	Count
Frequent Changes	2
Semi-frequent Changes	2
Infrequent Changes	4

Table 18: Question #17 — Degree of Changes to Database Objectives

Changes to Objectives in Database	Count
Major Changes	1
Semi-major Changes	2
Minor Changes	5

User Training and Support

Nearly everyone interviewed attended the training sessions (20 of 24; **Table 19**). Those who didn't attend did so because either they didn't think they would use the system, were too busy to attend, or didn't have enough notice to schedule them in (**Table 20**). Of those who attended the training sessions about half (12 of 20; **Table 21**) felt the training was adequate. Of those who felt the training was inadequate a number (5 of 8; **Table 22**) indicated that this was due to not having enough time to adequately learn the system. A few (3 of 8; **Table 22**) felt that it was too advanced — training on basic micro-computer skill should have been included with the sessions.

Everyone who used the system felt the user support was adequate (8 of 8; **Table 23**). Actually, based on some of their comments, the users felt the support to be more than merely adequate — it appears to have been excellent:

> "Was great!" — "I found ____ to be extremely helpful" — "definitely!" — "____ was always available when I need help"

Table 19: Question #6 — Did you attend the provided training sessions?

Attended Training	ELA	EMH	TMH	Total
Yes	12	5	3	20
No	2	2	0	4
Total	14	7	3	24

Table 20: Question #6 — Comment Summary of Reasons For Not Attending Training Sessions

Comment Category	Count
Not interested. / Didn't think I'd use the system.	2
Didn't hear about the training sessions soon enough	1
Didn't have the time to attend. / Too busy.	1

Table 21: Question #7 — Was adequate training provided?

Training	ELA	EMH	TMH	Total
Adequate	7	2	3	12
Inadequate	5	3	0	8
Total	12	5	3	20

Table 22: Question #7 — Comment Summary of Reasons Why Training was Inadequate

Comment Category	Count
Too short. / Not enough time to learn the system.	5
Too advanced. / Micro-computer introduction needed first.	3
Problem with training session — computers not working for part of the training session.	3
Should be held during school time, not after hours.	1
Should have time to work with my own data.	1

Table 23: Question #10 — Was adequate support provided when you had problems?

Adequate Support	ELA	EMH	TMH	Total
Yes	2	3	3	8
No	0	0	0	0
Total	2	3	3	8

Table 24: Question #8 — Did you use the provided user manual?

Use Manual	ELA	EMH	TMH	Total
Yes	2	3	3	8
No	0	0	0	0
Total	2	3	3	8

Future Use of the System

Although only one-third of the staff actually used the system (8 of 24; **Table 5**), all staff interviewed were asked if they would recommend continuing the use of the system. About three-quarters (19 of 24; **Table 25**) recommended continued use. No one in the other quarter were currently using the system nor did they have any plans to start.

A significant finding was that no one currently using the system recommended discontinuing its use — indicating that they were quite satisfied with it. In fact, of those currently using the system, most (5 of 8; **Table 27**) planned to increase their usage next year and the rest (3 of 8) said they would "stay the same". For the users that said they would stay about the same, two qualified their comments: "Depends on access to a computer", "If I had more access to a computer [I] would increase [my use]". The other person felt he/she was making "full use" of the system now so there was no way of increasing usage beyond that. Of those who weren't using the system, most (13 of 16; **Table 27**) indicated they would start using it next year. Only 3 indicated they had no plans to ever use the system:

> "I'm getting [close] to retiring... if I can get along without it until then, good." — "Only if I have to... the [hand-written] method I use now works fine"

Table 25: Question #12 — Would you recommend continuing use of the system?

Continue Use	ELA	EMH	TMH	Total
Yes	11	5	3	19
No	3	2		5
Total	14	7	3	24

Table 26: Question #12 — Comment Summary of Reasons for Not Continuing Use of the System

Comment Category	Count
Don't use it. / Don't plan on using it.	2
Like the old-fashion way. / Like to hand write comments.	2
Doesn't look like it's less work. If it doesn't save work then why use it.	1

Table 27: Question #5 — Do you anticipate your future use of the system to increase, decrease or stay about the same?

Usage	ELA	EMH	TMH	Total
Increase	11	5	2	18
Stay same	3	2	1	6
Decrease	0	0	0	0
Total	14	7	3	24

Suggestions for Improving the System

Probably the most interesting and valuable data collected from the interview sessions were suggestions on how the developed IPP system could be improved. This data came from questions 18, 19 and 20. Although each of these questions had a slightly different focus, the respondents didn't restrict their comments as narrowly. Therefore, the comment data was grouped and analyzed together in a single table (**Table 31**).

Slightly less than half of the interviewees (10 of 24; **Table 28**) thought some changes could be made to the system in order to make it more useful for them. The rest either thought what they had was adequate or couldn't think of any changes because they hadn't used it enough. Approximately the same proportion of those interviewees using the system (5 of 8; **Table 29**) thought more of the IPP process should be computerized. However, these numbers, although interesting, are not as important as the comments given and summarized in **Table 31**. It is important to note that there was no one change mentioned that stood out above the other comments in frequency. The most frequent comment arose only 5 times out of the 40 comments made. This indicates there was no

one issue that the respondents perceived to be lacking, or a common concern with the developed system.

The most frequent suggestion for improvement (5 times out of 40; **Table 31**) involved better access to computing resources. Although the frequency counts for this comment are not significantly different from a number of other comments in the table, the frequency counts do not do this comment justice. The frequency counts in the table only provide the reader with a two-dimensional picture. What is missing is the third-dimension — that of the emotional forcefulness with which this comment was made. During the interviews, computer access came across as quite a major concern. The importance of this comment also starts to gain more significance as one looks back at the responses to other questions. For example, this issue was the most frequently mentioned reason for not using the system in the first place (**Table 6**). In addition, even those who used the system emphasized computer access to be a problem which tended to reduce their frequency of use or act as a barrier to increased use ("If I had more access to a computer [I] would increase [my use]."). However, prior to implementation, steps were taken to provide what was thought to be adequate computer access. Every school had access to a number of computers designated solely for administrative use (see **Table 32**) and, in addition, many had labs of instructional computers (30+ machines) which could have been used to run the program. It appears that this access was not suitably adequate or convenient enough for the teaching staff involved in this study. A number of staff mentioned they would have liked to have had a computer right in their classroom or at home so they could have worked on the system at a time that was convenient for them.

Another comment mentioned frequently pertained to making the system easier to use and more straight forward and simpler to work with. Of the 4 people who made this comment, 3 did not use the system and all of them rated their comfort level with computers at the lowest possible point of the provided scale (5 - very uncomfortable; **Table 7**) — one even suggested he/she be rated past the end of the scale as a 6! Therefore, this may not be so much a specific criticism of this system rather more an admission that they find computers in general difficult to use. In contrast a number of other users, who rated themselves more comfortable with computers, commented on how easy it was to use:

> "[It's] so easy to use" — "Output looks good; easy to read; simple to look at and find information" — "I think it makes things so much easier" — "I found it fairly straight forward to understand"

It should be pointed out that some of the comments mentioned are in conflict with each other. For example, some teachers (4) felt the list of objectives should be expanded while others (2) felt it should be reduced in order to make a better system. This is normal for any development. There is no way all changes are going to please everyone. In the development of any system one constantly has to make trade-offs between conflicting features — a point which was discussed in detail in the previous chapter, **System Design, Development and Implementation**. Based on their comments, this point appears not to have been missed by the users:

> "Probably not able to generate a list [of features] that's OK for everyone." — "No one system can be designed to do more than about 70% of what one expects." — "I don't think a system can answer everything — I don't expect it to."

Table 28: Question #18 — Should any changes be made to the system in order to make it more useful for you?

Need Changes	ELA	EMH	TMH	Total
Yes	3	5	2	10
No	11	2	1	14
Total	14	7	3	24

Table 29: Question #19 — Should more of the IPP process be computerized?

More Computerized	ELA	EMH	TMH	Total
Yes	2	2	1	5
No		1	2	3
Total	2	3	3	8

Table 30: Question #20 — Are there any additional features/components, not already covered, you deem important for an ideal/future system?

Additional Features	ELA	EMH	TMH	Total
Yes	3	3	2	8
No	11	4	1	16
Total	14	7	3	24

Table 31: Question #18, 19, 20 — Comment Summary of Ways to Improve the System

Suggestions of Ways to Improve the System	Count
Need better access to a computer.	5
Information should follow student from one year to the next. / Have all student IPP data kept on-line for his/her entire school career.	4
Easier to use. / More straight forward and simpler to work with.	4
More and/or better training.	4
Revise/fix up existing objectives and expand (add more) objectives.	4
Faster program.	3
Have instructional/intervention strategies and/or example of evaluation criteria for each objectives.	3
Need place to put general comments on students (which are not on an objective by objective basis).	3
Need to reduce the number of objectives in the database.	2
Organize objectives by grade level.	2
More automatic IPP generation (talking to machine and/or press a single button to have it automatically generate IPPs).	2
Combine with the school's report card.	1
Way to add all objectives for a particular grade level.	1
Place to keep standardized student scores.	1
Automatically alert teacher if student is not meeting a certain % of his/her objectives.	1
Electronically send IPP data to others in the school system and/or province.	1
Comment Total:	40

Table 32: Administrative Computers per School

School	Teaching Staff	Administrative Computers
School A	69	26
School B	23	7
School C	35	16
School D	21	5
School E	20	5
School F	25	4
School G	11	4
School H	25	5
School I	8	3
School J	21	5
School K	26	5
School L	26	5
School M	21	4
School N	34	9

This table gives the number of computers available solely for administrative use. In addition, most of the schools also have a number of other computers (lab sets) available for instructional use which can be used for administrative purposes when not in use.

Chapter VI

Conclusion and Recommendations

Overview

The developed IPP management system (Wodelet, 1993) was used by about one-third of the interviewed special education staff in the XXX School District. Considering its use was on a purely voluntary basis and taking into account the extra work required to learn (and relearn) an evolving prototype, this is probably higher than one could normally expect. The prototype system appears to have been well received by the XXX staff as evidenced by the many positive post-implementation comments collected. In their suggestions for improvements there was no one change mentioned that stood out above the others, which indicates there was no one issue that the users perceived to be lacking, or a common concern with the developed system. Of the staff who did not use the system, many stated they plan to start using it the following school year. Inadequate computer access was probably the most serious barrier to use of the system. Not only did it prevent some staff from using the system, but it adversely effected the frequency of use of those who did. The following discussion will highlight this as well as other major findings of this study.

Existing IPP Management Systems

There appears to be a distinct lack of systems currently in use for the automated management of IPPs. Although the intent of this study was not to conduct an extensive or exhaustive review of all available systems, the investigation that was done uncovered relatively few systems. Furthermore, of those that were being used, none were very elaborate — they contained little or no specific customizations for the management of IPPs. The systems found were all basically word-processing documents with little or no specific IPP program code — they merely used the standard features provided with the application.

Therefore, not only does there appear to be a distinct lack of automated IPP systems, but what does exist appears to be very simplistic in nature. A more detailed and extensive study of automated IPP systems is required to support or refute these initial findings.

Prototyping Versus System Reviews

The prototyping process was far more useful than the system review for generating the list of features. The system review has helpful in refining and confirming the importance of some of the features identified during prototyping, but prototyping played a more important role in the generation of the list of features than did the system review.

It is important to note that the system review did not uncover any significant features that had not already been identified during the initial stage of prototyping. In fact, in some cases it took the prototyping process to "find" some of the features "missed" during the system review.

This is not to say that system reviews are not useful — they are — but they're most valuable when used in conjunction with a prototyping process. They should not be used as the sole mechanism for the generation of a list of features. The systems themselves should be considered as initial prototypes — with the information and ideas they generate feeding into the prototype developmental loop. The prototyping process seems to do a better job of eliciting user comments and ideas and hence the development of a better system.

Excel as a Prototyping Language

The choice of an appropriate prototyping language is a very important consideration for the development of any system — this study was no exception. Excel (Microsoft, 1992a), although it has many of the attributes of a 4th GL, has some serious drawbacks which keep it from being an adequate and robust prototyping language:

- It has no local variable name space. Much care is required not to use the same variable name in nested subroutines.
- There is no automatic deallocation of variable storage at the end of subroutines or programs.
- It lacks adequate debugging features.
- It has a number-driven argument passing mechanism which makes subroutine calls more difficult to understand.
- It has a relatively slow execution speed and consumes a significant portion of the available resources of a micro-computer. Speed of the final product becomes more of a problem as the size of the system grows.

This is not to say that it is totally unsuitable for the prototyping process because, to be fair, it does have some features which facilitate the quick development turnaround times and the creation of user friendly systems:

- It comes with a user friendly development environment (all window-based).
- It allows relatively novice users, through the use of the Dialog Editor, to easily construct window-based user interfaces with objects such as buttons, menus, list boxes, etc.
- It has an "English-like" command structure to help make the program code more self-documenting.
- It is an interpretive language so the effects of changes can be seen immediately instead of waiting for long save-compile-run cycles as exist for compiled developmental languages.
- The auto record feature helps novice users and developers alike to automatically "write" substantial sections of code quickly and easily.

Overall, on a scale of 1(excellent) to 5(poor), Excel would be rated at rated as a 4 (somewhat less than adequate) as a future choice for a prototyping language.

User / Developer Communication

Communication between users and developers is extremely important during the prototyping process. Electronic mail, although a potentially valuable tool for prototyping, was not extensively employed by users during this study to communicate with the developer. Since the e-mail system was available even during the running of the IPP system, it would have been easy for the user to simply mail off comments or suggestions as they occurred and return to exactly where they left off. However, most of the users distinctly preferred a more personable contact either through the use of telephone calls or in-person meetings. This may have been due to a lack of training on use of the e-mail system (it was not covered during any training sessions).

It is recommended that future studies take adequate steps to ensure open and frequent communication during the prototyping process. This could be done by providing basic e-mail training and ensuring the developer(s) are readily available for more personable/preferred forms of communication such as telephone calls and meetings.

Computer Literacy

Computer literacy seems to have been a major deterrent to use of the system. Approximately one-third of the respondents (7 of 24) respondents in this study rated themselves as being extremely uncomfortable in working with computers and nearly all of these individuals (6) did not use the system. While no one explicitly said, "*I'm computer illiterate and that's why I didn't use the system*", a number of comments

implied that they were afraid of computers or didn't have sufficient knowledge to use them easily or effectively.

> "I don't like computers at all; I'm very impatient working with computers; get quite uptight when working with them" — "looks like a lot more work", "I find computers a hassle I don't need" — "I'm getting [close] to retiring... If I can get along without it until then, good" — "I would do it by hand because that's easier and faster"

A computer literacy problem was also alluded to by some of the comments with respect to the adequacy of training given. Some respondents felt there should have been additional training given on basic micro-computer skills prior to the training sessions for the IPP system.

Computer literacy is currently a major problem for both school boards and industry in the implementation of computer technology into the work place. However, this problem should abate over time as today's more computer literate students move from the classroom into the work force. But for today's work force, computer literacy is a training problem — which will be discussed more fully in the following section on **Training**.

Training

Training is a very important part of the implementation of any system. It is important not to over-estimate, the basic micro-computer skill level of the staff to be trained. Although the training for the actual developed IPP system appeared to have been adequate, a number of complaints were raised about the inadequacy of training on the more basic/prerequisite skills — in particular micro-computer basics. Training on e-mail would have also be helpful to improve user/developer communication during the prototyping process (as discussed in more detail in **User / Developer Communication**).

This study found that users would have preferred some additional follow-up training sessions to allow them time to develop IPPs using their own student data. These additional training sessions should also have a resource person available to assist them with any problems which may arise.

Computer Access

One of the major problems uncovered by this study concerned the lack of access to computing resources. This was the most frequent reason given for lack of use of the system and one of the most frequent suggestions made regarding ways to improve the system. It was even cited as a factor effecting the frequency of system use. A number of users stressed that they would have liked to use the system more but had problems accessing a computer regularly or access wasn't convenient enough to enable them to use the system on a frequent basis.

Future implementors of automated IPP systems must be cognizant that the issue of adequate computer access needs to be addressed before a successful system implementation can occur. It is not good enough just to look at the numbers of computers available in a school (as this study did prior to implementation) — their accessibility must also be considered. Although every school in the XXX School District had a number of computers available, availability did not mean they are readily/conveniently accessible. Some computers were locked up after hours, others did not have the required software installed, a few had inadequate memory to run the IPP system or, most frequently, were used for other purposes during the periods of time when teachers wanted to access them.

Perhaps inadequate computer access could help explain the apparent lack of automated IPP systems in schools. This should be investigated further by future studies in this area.

Recommendations for Future Development

The system developed by this study should not be considered a completed system. Although it is fully functional system and can be used as a stand-alone system, it was designed to exist and be used in the context of a larger framework which collects and maintains other needed IPP information (such as the information collected by the form shown in **Appendix I**). Ideally the developed system should be more tightly coupled with all other parts of the IPP process to provide a more integrated and complete solution to providing individualized plans for students.

It is recommended that the prototype process be continued with the system for at least one additional complete school year. A complete school year is important to ensure all parts of the IPP process — from start to finish — could be exercised. An additional year would help produce a better system because of the projected greater numbers of staff involved. While approximately one-third of the teaching staff used the system during the first year of development, a larger number indicated they would not be using it until the following year. Not only would this increased number result in a greater input of suggestions and ideas into the prototyping process (and hence a better system) but it would also elicit responses from a slightly different user group. Users closely involved with the development of any systems tend to become "ego-involved". They feel this is *their system* and tend to become more resistant to change. New users would help bring a more disinterested and fresh perspective to the next stage of evolution. Furthermore, one finding of this study was that the existing users rated themselves slightly more computer literate/comfortable than the non-users. As a result the comments and suggestions of the existing users (which were used to develop the prototype) may not be totally representative of the other staff in the XXX School District. Therefore, the comments and suggestions from less computer literate users could serve to develop an even more functional and easier to use system — at least from the aspect of less literate users.

The second prototyping phase will likely not need to have as an intensive and rapid design–development loop as occurred in the initial development phase of this study. The reason for this is that the more stable and robust a system becomes the less likely one is to find serious flaws or omissions which need to be quickly fixed in order to provide a functional user system. However, even in mature systems, although the design–development loop may slow it will likely never stop unless the system become truly non-

functional. This is because over time users' needs and requirements change to keep pace with changing programs and policies. Those systems which do not keep pace with these changing needs are destined, through eventual obsolescence, to fade into obscurity.

Bibliography

Alberta Education, (1987). *Visions 2000: A vision of educational technology in Alberta by the year 2000.* Edmonton, Alberta: Student Programs and Evaluation Division.

Alberta Education, (1989). *Program policy manual.* Alberta Education, Edmonton, Alberta.

Alberta Education, (1990). *School Act.* Alberta Education, Edmonton, Alberta.

Alberta Education Response Center, (1989). *Individualized program plans: A reference for teachers.* Alberta Education, Edmonton, Alberta.

Baker, F. (1981). Computer managed instruction: A context for computer based instruction. In H.F. O'Neil (Ed.), *Computer based instruction: A state-of-the-art assessment* (pp. 23-64). Toronto: Academic Press.

Bangert-Drowns, R.L., Kulik, J.A. & Kulik, C.C. (1985). Effectiveness of computer-based education in secondary schools. *Journal of Computer-Based Instruction.* 12, 59-68.

Bloom, B. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher*, 13(6), 4-16.

Borg, W.R. & Gall, M.D. (1989). *Educational Research.* New York: Longman.

Brown, N.P. (1982). CAMEO: Computer-assisted management of educational objectives. *Exceptional Children*, 49(2), 151-153.

Bureau of Education for the Handicapped. (1979). *Progress toward a free, appropriate public education: A report to congress on the implementation of public law 94-142: The education for all handicapped children act.* Washington D.C.

Carrier, C., Davidson, G. & Williams, M. (1985). The selection of instructional options in a computer-based coordinate concept lesson, *Educational Communication and Technology Journal*, 33, 199-212

Claris (1988). Apple*Works* [Computer program]. Claris Corporation, 5201 Patrick Henry Drive, PO Box 58168, Santa Clara, California, 95054-8168, U.S.A..

Clark, R.E. (1983). Reconsidering research on learning from media. *Review of Educational Research*, 53(4), 445-460.

Clark, R.E. & Salomon, G. (1986). Media in teaching. In M. Wittrock (Ed.), *Handbook of Research on Teaching, third edition.* New York: Macmillan.

Clark, R.E. & Sugrue, B.M. (1988). Research on instructional media, 1978-1988. In D. Ely (Ed.), *Educational Technology and Media Yearbook*, (pp. 19-36) Englewood, CO: Libraries Unlimited.

David, J.L. (1991). Restructuring and technology: Partners in change. *Phi Delta Kappan*, 73(1), 37-82.

Davies, I.K. (1973). *Competency based learning: Technology, management, and design.* New York: McGraw-Hill.

Dewey, J. (1916). *Democracy and education.* New York, NY: Macmillan.

Dewey, J. (1938). *Experience and education.* New York, NY: Macmillan.

Dudley-Marling, C. (1985). Perceptions of the usefulness of the IEP by teachers of learning disabilities and emotionally disturbed children. *Psychology in the Schools*, 22, 65-67.

Dick, W. & Carey, L. (1990). *The systematic design of instruction - 3rd ed.* Illinois: Scott, Foresman/Little, Brown.

Dwyer, F.M. (1978). *Strategies for improving visual learning.* State College, PA: Learning Services.

Enell, N.C. (1984). *A cost comparison of preparing special education individualized programs with and without computer assistance.* American Research Association, New Orleans.

Enell, N.C. & Barrick, S.W. (1983). *An examination of the relative efficiency and usefulness of computer-assisted individualized education programs.* California State Department of Education.

Gentry, C.G. (1991). Educational technology: A question of meaning. In Anglin, Gary J. (Ed.), *Instructional Technology: Past, Present, and Future.* Englewood, CO: Libraries Unlimited.

Gentry, C.G. & Csete J. (1991). Educational technology in the 1990s. In Anglin, Gary J. (Ed.), *Instructional Technology: Past Present, and Future.* Englewood, CO: Libraries Unlimited.

Jenkins, M.W. (1987). Effect of a computerized individual education program (IEP) writer on time savings and quality. *Journal of Special Education Technology*, 8(3), 55-66.

Kolb, D.A. (1984). *Experiential learning: Experience as the source of learning and development.* Englewood Cliffs, NJ: Prentice-Hall.

Kulik, J.A., Bangert, R.L. & Williams, G.W. (1983). Effects of computer based teaching of secondary school students. *Journal of Educational Psychology*, 75, 19-26.

Kulik, J.A., Kulik, C.C. & Cohen, P.A. (1980). Effectiveness of computer-based college teaching: A meta-analysis of findings. *Review of Educational Research*, 50, 525-544.

La Follette J.J. (1992). Instructional technology and teacher education. Canadian *Journal of Educational Communication*, 21(2), 109-122.

Lange, P.C. (1967). Introduction: Issues and problems. In Lange, P.C. (Ed.), *Programmed Instruction: The Sixty-Sixth Yearbook of the National Society for the Study of Education*. Chicago, Illinois: University of Chicago Press.

Lee, W.C. (1990). *The effectiveness of computer-assisted instruction and computer programming in elementary and secondary mathematics*. Unpublished doctoral dissertation, University of Massachusetts.

Lunenburg. (1993). *IEP Database* [Computer program]. Lunenburg County District School Board, P.O. Box 380, 66 Pleasant Street, Bridgewater, Nova Scotia, B4V 2W9.

MicroMedia. (1993). *IEPWorks* [Computer program]. MicroMedia Publishing, 6 Arrow Road, Ramsey, NJ 07446.

Microsoft. (1990). *Works* [Computer program]. Microsoft Corporation, One Microsoft Way, Redmond, Washington, 98052-6399, U.S.A..

Microsoft. (1992a). *Excel* [Computer program]. Microsoft Corporation, One Microsoft Way, Redmond, Washington, 98052-6399, U.S.A..

Microsoft. (1992b). *Word* [Computer program]. Microsoft Corporation, One Microsoft Way, Redmond, Washington, 98052-6399, U.S.A..

Martin, J.E. (1984). *An information systems manifesto*. Englewood Cliffs, NJ: Prentice-Hall.

Martin, J.E. (1989). *Information engineering: A trilogy*. Englewood Cliffs, NJ: Prentice-Hall.

McClure C.A. (1992). *The quality of computerized versus non-computerized individual education plans*. Unpublished master's thesis, San Diego State University, San Diego.

Morrison, G.R., Ross, S.M. & O'Dell, J.K. (1991), Applications of research to the design of computer-based instruction. In Anglin, Gary J. (Ed.), *Instructional Technology: Past Present, and Future*. Englewood, CO: Libraries Unlimited.

Niemiec, R., Samson, G., Weinstein, T. & Walberg, H.J. (1987). The effects of computer based instruction in elementary schools: A quantitative synthesis. *Journal of Research on Computing in Education*, 20, 85-103.

Palfreman J. (Producer) & Apsell P.S. (Director). (1992). *The machine that changed the world* [Videotape]. Boston, MA: WGBH Educational Foundation.

Reiser, Robert A. & Salisbury, David F. (1991). Instructional technology and public education in the United States: The next decade. In Anglin, Gary J. (Ed.), *Instructional Technology: Past, Present, and Future*. Englewood, CO: Libraries Unlimited.

Reynolds, C.R. (1988). Putting the individual into aptitude treatment interaction. *Exceptional Children*, 54, 324-331.

Ross S.M. & Morrison G.R. (1989). In search of a happy medium in instructional technology research: Issues concerning external validity, media replications, and learner control. *Educational Technology Research and Development*, 37(1), 19-33.

Ryan, L.B. & Rucker, C.N. (1986). Computerized vs. non-computerized individualized education programs: Teachers' attitudes, time and cost. *Journal of Special Education Technology*, 8(1), 5-12.

Ryan, L.B. & Rucker, C.N. (1991). The development and validation of a measure of special education teacher's attitudes toward the individualized educational program. *Educational and Psychological Measurement*, 51, 887-883.

Schultz, J.B. & Turnbull, A.P. (1984). *Mainstreaming handicapped students: A guide for classroom teachers (2nd ed.)*. Boston : Allyn and Bacon Inc.

Smith, S.W. (1990). Individualized education programs in special education: From intent to acquiescence. *Exceptional Children*, 57(1), 6-14.

Sudman, S. & Bradburn N.M. (1983). *Asking Questions*. San Francisco: Jossey-Bass.

Szabo, M. & Montgomerie, T.C. (1992). Two decades of research on computer-managed instruction. *Journal of Research on Computing in Education*, 25(1), 113-133.

Tennyson, R.D. (1980). Instructional control strategies and content structure as design variables in concept acquisition using computer-based instruction. *Journal of Educational Psychology*, 72, 525-532.

Tennyson, R.D. (1981). Use of adaptive information for advisement in learning concepts and rules using computer assisted instruction. *American Educational Research Journal*, 18, 425-438.

University of Alberta (1991). *University standards for the protection of human research participants*. Edmonton, Alberta.

Wittrock, M.C. (1974). Learning as a generative process. *Education Psychologist*, 11(2), 87-95.

Wodelet, D.L. (1993). *IPP System* [Computer program]. Strathcona County Schools, 2001 Sherwood Drive, Sherwood Park, Alberta, Canada, T8A 3W7.

XXX School District. (1992). *Special Needs Handbook*. XXX's City, Alberta, Canada.

Appendices

Appendix A

Consent Form

The contents of the consent form were explained to the interviewee and signed prior to the interview process.

Consent Form

Thesis Title: **The Development and Validation of a Computerized IPP Management System through Prototyping**

Researcher: **D. L. Wodelet**
University of Alberta
Faculty of Education
Department of Adult Career and Technology

This is to certify that I agree to participate in the above study. Having been contacted by the researcher, a graduate student in the Department of Adult Career and Technology, I understand that:

1. The purpose of this study is to design, develop, implement and evaluate an automated/computerized system for the management of students in individualized learning environments.
2. My name will not be disclosed at any time during this study or used in the resulting thesis.
3. Any information I provide to the researcher will be kept confidential and used solely for the purposes of this research study.
4. I am participating in this study on a purely voluntary basis. Therefore, I have the right to quit or refuse to participate at any time.
5. The results of the study will be made available to me if I so request.
6. I have been fully informed as to the nature of the study and my involvement in it.
7. The thesis this study leads to will be available for examination at the University of Alberta Library.

Signature of participant _____ **Date** _____

Appendix B

Pre-Interview / Contact Guide

This guide outlines the steps to be followed and topics to be covered prior to the main interviews and data collection sessions outlined in **Appendix C** and **Appendix D**. The other interview guides in these appendices depends on this being done prior to their use. This guide was created using many of the suggestions found in Borg & Gall (1989) and Sudman & Bradburn (1983) regarding data collection using an interview process. Once constructed, a practice run using this guide was carried out on a few subjects to isolate problem areas and help the researcher gain experience with the pre-interview process.

- Introduce myself. Briefly indicate why I am contacting them. Confirm if this is a convenient time to discuss this study — if not reschedule.
- Give overview of study stressing its significance to help encourage them to participate. Offer a summary of the finished thesis for their participation.
- Give interviewee some idea of the anticipated time commitment required should he/she agree to participate.
- Tell the interviewee how confidentiality will be maintained. Only the researcher will have access to the data from the interview. Data will not be kept on any file server or machine with public access. All personal notes will be destroyed at the end of study.
- Inform interviewees will have the opportunity to review their transcribed comments and make necessary corrections before their comments are used. A consent form (**Appendix A**) will need to be signed to allow use of their comments.
- Stress that complete anonymity will be maintained — their names will not be used in any written summary or verbal discussions.
- Try to setup a convenient time for the interview/next contact.

Appendix C

Interview Guide: Determining System Features

This interview guide was created using many of the suggestions found in Borg & Gall (1989) and Sudman & Bradburn (1983) regarding data collection using an interview process. This guide was used to determine what features/components a computerized system for the management of students in individualized program plans should contain. This is essentially an unstructured interview since this is the best technique to collect the "open-ended" data required. This guide was used to help structure a number of informal meetings and brainstorming sessions held with teachers and administrative support staff involved with the delivery of individualized programs. Notes from these informal sessions were used to generate an list of features/components considered important for the automated management of IPPs.

- This guide assumes the participants to be interviewed have been previously contacted using the *Pre-Interview / Contact Guide* in **Appendix B**.
- Introduce myself and the study being done to help refresh their memory of the things covered in pre-interview contact.
- Confirm that this is a convenient time for the interview — if not reschedule.
- Confirm the anticipated time required to complete this interview to ensure we'll have enough time to complete it — if not reschedule.
- Reaffirm that confidentiality will be maintained.
- Provide overview of the study for the interviewee and the goal of this particular interview — to determine what features/components are important for the automated management of IPPs.
- Start by allowing interviewees to talk about themselves to help them gain some comfort with the interview process. For example, may want to have the interviewee talk about their past experiences in education, university attended, their family, hobbies, etc.
- Have the interviewee go over what things they currently do and the tools they currently use when creating and modifying student IPPs. This will get the interviewee thinking about the IPP process and prepare him/her for the next question — what features/components they think are important for managing student IPPs.
- Ask what features/components they think are important for managing student IPPs. Another way to phrase this question would be to ask what features they would consider to be in an ideal system to manage student IPPs.
- Bring closure to the interview: Review the goals of interview and let the interviewee bring closure by asking if there is anything more they would like to add

and/or asking if there was something they felt was missed which should be mentioned.

- Thank them for their participation. Let them know their comments and insights were greatly appreciated and will be very useful for this research.

Appendix D

Interview Guide: Users' Views and Perceptions

This interview guide was created using many of the suggestions found in Borg & Gall (1989) and Sudman & Bradburn (1983) regarding data collection using an interview process. This guide was used to determine the views and perceptions of users after the implementation of a computerized system for the management of students in individualized program plans. This interview contains both structured and unstructured components. Once constructed, a practice run using this guide was carried out on a few subjects to isolate problem areas and help the researcher gain experience with the interview process.

- This guide assumes the experts to be interviewed have been previously contacted using the *Pre-Interview / Contact Guide* in **Appendix B**.
- Introduce myself and the study being done to help refresh their memory of the things covered in pre-interview contact.
- Confirm that this is a convenient time for the interview — if not reschedule.
- Confirm the anticipated time required to complete this interview to ensure we'll have enough time to complete it — if not reschedule.
- Reaffirm that confidentiality will be maintained.
- Provide overview of the study for the interviewee and the goal of this particular interview — to determine the user's views and perceptions on the use of the computerized IPP system.
- Allow interviewee to talk about him/herself to help gain comfort with the interview.
- Ask the structured questions. These questions were expanded upon and refined through information collected from both the "System Features" interview and development and implantation process. Prior to asking any questions, show a sample screen from the IPP system so there is no doubt what computerized system we will be talking about. Note: participants who did not use the computerized IPP system (answered "no" to question #1) were only asked those questions numbered with an outline style.

  1) Did you use the IPP computerized management system? If not why not? If no then only ask those questions numbered with an outline style.
  2) When did you start using the IPP system?
  3) How often do you use the actual computerized system (daily, weekly, monthly, every reporting period, etc.)?

4) How often do you use the data sheets provided by the system (daily, weekly, monthly, every reporting period, etc.)?

5) Do you anticipate your future use of the system to increase, decrease or stay about the same?

6) Did you attend the provided training sessions? If not, why not? If answer is NO skip the next question.

7) Was adequate training provided? If not, in what way was it inadequate?

8) Did you use the provided user manual (show a copy of the user manual to refresh their memory)? If not, why not?

9) Once you learned how to use the computerized IPP system did it save time?

10) Was adequate support provided when you had problems?

11) How would you rate the system overall on a scale from 1 to 5 (where 1 is exceptionally useful and 5 is not useful at all)?

12) Would you recommend continuing use of the system? If no, why not?

13) Of the students you need to write IPPs for, approximately what percentage these students did you put on the system?

14) How many students do you write IPPs for (over the period of one school year)?

15) On the average, how many IPP objectives do you maintain on a student (over the period of one school year)?

16) How would you rate your comfort level in working with computers on a scale from 1 to 5 (where 1 is very comfortable and 5 very uncomfortable)?

17) Did you need to change the objectives provided in the database? If so, was this frequently done? Were these minor or major changes?

18) Should any changes be made to the system in order to make it more useful for you? If so, what changes?

19) Should more of the IPP process be computerized? If so, what part?

20) Are there any additional features/components, not already covered, you deem important for an ideal/future system? If so, what?

- Bring closure to the interview. Review the goals of interview and then let the interviewee bring closure by asking if there is anything more they would like to add or if there was something they felt was missed which should be mentioned.
- Reconfirm that they will have the opportunity to read the transcribed comments and make necessary corrections before their comments are used. Arrange a time to contact them again in case clarification is needed when transcribing notes.
- Thank them for their participation. Let them know their comments and insights were greatly appreciated and will be very useful for this research.

Appendix E

Program Listing

This appendix contains the program listing for the IPP computerized system. Some of the program lines have been truncated in order to fit on a single page. The complete detail can be found in the source code for "IPP Macros". A diskette containing complete program along with the source code has been included in the enclosure at the back of this thesis.

names	commands	comments
	Summary_Information	
Title:	IPP Macros	
arVersion:	1.8	
arAuthor	D. L. Wodelet	
Creation Date:	"September1992"	

	auto_open	*Only called when this file opene*
arThisMacroFileNa	=GET.CELL(32,A9)	*get name of this document*
	=IF(GET.WINDOW(7,arThisMacroFileName),UNHIDE(	*if window is hidden unhide it*
	=FULL(TRUE)	*reverse size change by auto_clo*
	;;;=WINDOW.SIZE(560,320,arThisMacroFileName)	
	=RETURN()	

	auto_close	
	=ECHO(FALSE)	*no screen update for speed*
	=IF(GET.DOCUMENT(4,arThisMacroFileName))	*has this document been change*
	= IF(GET.WINDOW(7,arThisMacroFileName),,WIND	*if window not hidden (otherwise*
	= ACTIVATE(arThisMacroFileName)	*make this the active window*
	= SELECT(arAuthor)	*So our names come up on open*
	= SAVE()	*save it so user isn't asked*
	=END.IF()	
	=RETURN()	

	IPPAuto_Open	*run by Auto_Open in Template*
	;;;ECHO(FALSE)	*DON'T turn updates off - just ca*
	=IF(NOT(GET.WINDOW(7,arThisMacroFileName)))	*if window not hidden then hide*
	= WINDOW.SIZE(60,63,arThisMacroFileName)	*make small so doesn't mess up*
	= ACTIVATE(arThisMacroFileName)	*make this the active window*
	= HIDE()	*hide the window*
	=END.IF()	
	=IF(CreateNewPlan(FALSE),IPPNewPlan())	*Create a new plan if one doesn'*
	=RETURN()	

	IPPHelp	*give help on IPP buttons*
	=ALERT("-- Program Version "&arVersion&" -- Th	
	=RETURN()	

	IPPHelpFormat	*give help on IPP Format buttons*
	=ALERT("These buttons allow you to reformat the	
	=RETURN()	

	IPPHelpIndent	*give help on IPP Indent buttons*
	=ALERT("These buttons allow you to ADD or REMO	
	=RETURN()	

	IPPNewPlan	Creates a NEW IPP Plan/worksh
	=ECHO(FALSE)	no screen update for speed
arNewStudentNar	=INPUT("Enter Student's Name:",2,"Create New IPI	get student's name
	=IF(arNewStudentName,,HALT())	quit if Cancel button pressed
	=CreateNewPlan(TRUE)	Create the new plan
	=RETURN()	

	CreateNewPlan	Checks to see if Current Plan is
	=ARGUMENT("aCreateNewPlan",4)	See if I should create a new pla
	=RESULT(4)	routine returns a logical
	aCellRefPrefix="R"&ROW(!Print_Titles)&"C"	get reference for row "IPP for..."
	aCellRef=aCellRefPrefix&COLUMN(!Print_Titles)	create reference for where the "
arThisPlanIsInUse	=TRIM(TEXTREF("!"&aCellRef))<>"IPP for"	
	=IF(aCreateNewPlan)	
	= IF(arThisPlanIsInUse)	does this document have data fo
	aTemplateFileSpec=GET.DOCUMENT(2,arThisI	get filespecs for this macro file
	= FOR("aidx",LEN(aTemplateFileSpec),0,-1)	loop to find last ":"
	= IF(MID(aTemplateFileSpec,aidx,1)=":",BREA	quit when 1st colon is found
	= NEXT()	
	aTemplateFileSpec=MID(aTemplateFileSpec,1	get filespecs for template file (lo
	= SET.NAME("aidx")	remove temorary variable
	= ERROR(0)	I will trap the open error and HA
	= IF(ISERROR(OPEN(aTemplateFileSpec,0,,,,,TRU	open file
	= ALERT("Can't find IPP Template file '"&aTe	
	= HALT()	
	= END.IF()	
	= ERROR(1)	Turn error checking back on
	= SET.NAME("aTemplateFileSpec")	remove temorary variables
	= END.IF()	
	= FORMULA("IPP for "&arNewStudentName,aCellR	Add the "IPP for...." text
	aCellRef=aCellRefPrefix&GET.DOCUMENT(12)	create reference for where the "
	aYearStart=YEAR(NOW())-IF(MONTH(NOW())>(	if beween June-Dec then use cu
	= FORMULA(TEXT(aYearStart,"0")&"/"&RIGHT(TE	Add the current school year
	= COLUMN.WIDTH(0.1,aCellRef)	set a with just big enough so the
	= SET.NAME("aActCell",ACTIVE.CELL())	save current active cell
	= SELECT(aCellRef)	
	= ALIGNMENT(4,FALSE)	set right justify
	= SELECT(aActCell)	restore active cell
	= SET.NAME("aActCell")	remove temorary variables
	= SET.NAME("aYearStart")	
	= SELECT("R"&ROW(!Print_Titles)+ROWS(!Print_	move to 1st row past titles in th
	= SetPrint_Area()	reset Print_Area in case it is me
	=END.IF()	
	=SET.NAME("aCellRefPrefix")	remove temorary variables
	=SET.NAME("aCellRef")	
	=SET.NAME("aCreateNewPlan")	

	=RETURN(NOT(arThisPlanIsInUse))	

	IPPDelete	*Deletes IPP in current row*
	=ECHO(FALSE)	*no screen update for speed*
	=EDIT.DELETE(3)	*delete current row*
	=RenumberObjectives()	*renumber the objectives*
	=RETURN()	

	IPPInsert	*Insert new IPP above current ro*
	=AddNewIPP(TRUE)	*insert new row before adding IP*
	=RETURN()	

	IPPChange	*Changes IPP in current row*
	=AddNewIPP(FALSE)	*Change/replace IPP in current r*
	=RETURN()	

	IPPViewPrintDatabase	*Views/Prints the IPP database*
	=RESULT(4)	*return logical - false if cancel bu*
arViewDatabase	=ALERT("The "&CHOOSE(ud.ObjDB,"TMH","EMH",	
	=IF(arViewDatabase)	*was Cancel button pressed?*
	= GetDataFile()	*no, load objective database file*
	= IF(GET.WINDOW(7,IPPDataFile),UNHIDE(IPPData	*if window is hidden unhide it*
	= ACTIVATE(IPPDataFile)	*make this the active window*
	=END.IF()	
	=RETURN(arViewDatabase)	

	AddNewIPP	*Prompt and adds new IPP*
	=ARGUMENT("aInsertNewRow",4)	*See if I should add new row on a*
	=ECHO(FALSE)	*no screen update for speed*
arGetObjective	=DIALOG.BOX(udGetIPPDialog)	*Get IPPcode and othe paramete*
	=IF(NOT(arGetObjective))	*was Cancel button pressed?*
	;;;;FORMULA(FALSE,ud.AddRepeating)	*don't mess with default value -*
	= RETURN()	*quit*
	=ELSE.IF(arGetObjective=ROW(ud.ViewPrintDatabas	*was the View/Print button press*
	= IF(IPPViewPrintDatabase(),,GOTO(arGetObjectiv	*Views/Print the IPP database, g*
	= RETURN()	
	=END.IF()	
	=MESSAGE(1,"Looking for IPP code "&ud.IPPcode	
	=GetDataFile()	*Load File with IPP Objectives*
arIPPMatchIdx	=MATCH(ud.IPPcode,TEXTREF(""&IPPDataFile&"'!IF	*lookup IPPcode in table*
	=IF(ISERROR(arIPPMatchIdx))	
	= ALERT("Invalid IPP objective code: "&ud.IPPco	
	= GOTO(arGetObjective)	*get another objective*
	=END.IF()	
	=IF(ROW(ACTIVE.CELL())<ROW(!Print_Titles)+ROV	*is current active cell in the print*
	= SELECT("R"&ROW(!Print_Titles)+ROWS(!Print_	*move to 1st row past titles in t*

	aInsertNewRow=TRUE	*force the insert of a new row sc*
	=END.IF()	
	=IF(aInsertNewRow,INSERT(3))	*insert new row if necessary*
	=AddObjective(arIPPMatchIdx)	*add the IPP Objective*
	=RenumberObjectives()	*renumber the objectives*
	=SELECT("R[1]C")	*Move down to next row*
	=SetPrint_Area()	*reset the print area*
	=IF(ud.AddRepeating)	*does user want a repeating add'*
	= ECHO(TRUE)	*Show user what's been updated*
	= ECHO(FALSE)	
	= GOTO(arGetObjective)	*loop back to get another object*
	=END.IF()	
	=MESSAGE(FALSE)	*Remove/clear message area*
	=RETURN()	

	AddObjective	*Adds the IPP Objective indexed*
	=ARGUMENT("aRowOffset",1)	*Get row offset of IPP objective in*
	aDataFileRef="'"&IPPDataFile&"'!R"&ROW(OFFSET(	*build string to reference the Object*
	aColOffset=4	*try the objective column 1st (th*
	=WHILE(ISBLANK(TEXTREF(aDataFileRef&aColOffs(	*keep on looping backwards while*
	aColOffset=aColOffset-1	*decrement column to try*
	= IF(aColOffset=0)	
	= ALERT("Couldn't find a non-blank cell on row	*abort if no blank column found*
	= HALT()	
	= END.IF()	
	=NEXT()	
	aDataFileRef=aDataFileRef&aColOffset	*construct full reference*
	=SET.NAME("aActCell",ACTIVE.CELL())	*save current active cell*
	=SELECT("RC1:RC"&COLUMN(!Col_Objectives))	*Select from C1:<to the objectiv*
	=CLEAR(3)	*Clear the area*
	=SELECT("RC"&aColOffset-1)	*select the cell where the object*
	=FORMULA(TEXTREF(aDataFileRef),ACTIVE.CELL())	*Insert objective into cell*
	aAddBoldedText=GET.CELL(20,TEXTREF(aDataFileF	*see if text is bolded*
	=FORMAT.FONT(,GET.CELL(19,TEXTREF(aDataFile	*set bolding/fontsize,etc same a*
	=ALIGNMENT(1,NOT(aAddBoldedText))	*turn on wrap if NOT bolded (bol*
	=ROW.HEIGHT(,ACTIVE.CELL(),,3)	*re-wrap the row*
	=IF(aColOffset=2)	*is this a major headings?*
	= ROW.HEIGHT(acMajorHeadingRowHeight)	*make row height bigger*
	=END.IF()	
	=SELECT("RC"&COLUMN(!Col_Objectives)&":RC"&G	*select all cells in current row fro*
	=BORDER(,,1,,1)	*set a right and bottom border o*
	=SELECT("RC1:RC"&COLUMN(!Col_Objectives)-1)	*select all cells in current row fro*
	=BORDER(,,0,,1)	*set a bottom border on these c*
	=SELECT(aActCell)	*restore active cell*
	=SET.NAME("aActCell")	*remove temorary variables*
	=SET.NAME("aRowOffset")	

	=SET.NAME("aColOffset") =SET.NAME("aDataFileRef") =SET.NAME("aAddBoldedText") =RETURN()	

	RenumberObjectives	
	=SET.NAME("aActCell",ACTIVE.CELL()) =SELECT("RC"&COLUMN(!Col_Objectives)) =IF(ISBLANK(ACTIVE.CELL())) = SELECT("RC[-1]") = IF(ISNUMBER(ACTIVE.CELL()),CLEAR(3)) =ELSE() aTheNumB4=IF(ROW(ACTIVE.CELL())=1,0,OFFS aTheNumB4=IF(ISNUMBER(aTheNumB4),aTheN aTheNumB4=IF(aTheNumB4<0,0,aTheNumB4) = FORMULA(aTheNumB4,OFFSET(ACTIVE.CELL() = FOR.CELL("aTheNumB4Ref",OFFSET(ACTIVE.CE = IF(ISNUMBER(aTheNumB4Ref),,BREAK()) aTheNumB4=aTheNumB4+1 = FORMULA(aTheNumB4,aTheNumB4Ref) = SELECT(aTheNumB4Ref) = ALIGNMENT(1,TRUE) = FORMAT.FONT(,,FALSE) = FORMAT.NUMBER("##.") = NEXT() = SET.NAME("aTheNumB4Ref") = SET.NAME("aTheNumB4") =END.IF() =SELECT(aActCell) =SET.NAME("aActCell") =RETURN()	*Renumbers the IPP Objectives* *save current active cell* *goto the column the objectives* *if the Col_Objectives is blank th* *Go back one column to number* *if it contains a numberclear it s* *Col_Objectieves is NOT blank* *get the number from the row ab* *if it wasn't a number then make* *make sure number is >= 0* *set 1st number* *Get the offset of the start of th* *quit on 1st non-numeric cell* *increment number* *insert the number* *select need for following ALIGNM* *Turn on wrap so number goes t* *remove bolding from any previo* *format the cell* *remove temporary variables* *restore active cell* *remove temporary variables*

	IPPCheckList	
	=ECHO(FALSE) =MESSAGE(1,"Formating for IPP Checklist...") =COLUMN.WIDTH(,!Col_ExpectedAchieveDate,,2) =COLUMN.WIDTH(,!Col_Achievement,,1) =COLUMN.WIDTH(,!Col_CheckList,,2) =COLUMN.WIDTH("C"&COLUMN(!Col_CheckList),,1 =COLUMN.WIDTH(,!Col_ReportCard,,1) =COLUMN.WIDTH(,!Col_ProgramPlan,,1) =SetObjColWidth() =MESSAGE(FALSE) =RETURN()	*no screen update for speed* *unhide* *hide* *unhide* *hide 1st column - only there so* *hide* *hide* *Set width and rewrap Obj Colum* *Remove/clear message area*

	IPPReportCard	

	=ECHO(FALSE)	*no screen update for speed*
	=MESSAGE(1,"Formating for IPP Report Card...")	
	=COLUMN.WIDTH(,!Col_ExpectedAchieveDate,,2)	*unhide*
	=COLUMN.WIDTH(,!Col_Achievement,,1)	*hide*
	=COLUMN.WIDTH(,!Col_CheckList,,1)	*hide*
	=COLUMN.WIDTH(,!Col_ReportCard,,2)	*unhide*
	=COLUMN.WIDTH(,"C"&COLUMN(!Col_ReportCard),	*hide 1st column - only there so*
	=COLUMN.WIDTH(,!Col_ProgramPlan,,1)	*hide*
	=SetObjColWidth()	*Set width and rewrap Obj Colum*
	=MESSAGE(FALSE)	*Remove/clear message area*
	=RETURN()	

	IPPAchievement	
	=ECHO(FALSE)	*no screen update for speed*
	=MESSAGE(1,"Formating for IPP Achievement...")	
	=COLUMN.WIDTH(,!Col_ExpectedAchieveDate,,2)	*unhide*
	=COLUMN.WIDTH(,!Col_Achievement,,2)	*unhide*
	=COLUMN.WIDTH(,"C"&COLUMN(!Col_Achievemen	*hide 1st column - only there so*
	=COLUMN.WIDTH(,!Col_CheckList,,1)	*hide*
	=COLUMN.WIDTH(,!Col_ReportCard,,1)	*hide*
	=COLUMN.WIDTH(,!Col_ProgramPlan,,1)	*hide*
	=SetObjColWidth()	*Set width and rewrap Obj Colum*
	=MESSAGE(FALSE)	*Remove/clear message area*
	=RETURN()	

	IPPProgramPlan	
	=ECHO(FALSE)	*no screen update for speed*
	=MESSAGE(1,"Formating for IPP Program Plan...")	
	=COLUMN.WIDTH(,!Col_ExpectedAchieveDate,,1)	*hide*
	=COLUMN.WIDTH(,!Col_Achievement,,1)	*hide*
	=COLUMN.WIDTH(,!Col_CheckList,,1)	*hide*
	=COLUMN.WIDTH(,!Col_ReportCard,,1)	*hide*
	=COLUMN.WIDTH(,!Col_ProgramPlan,,2)	*unhide*
	=COLUMN.WIDTH(,"C"&COLUMN(!Col_ProgramPlan	*hide 1st column - only there so*
	=SetObjColWidth()	*Set width and rewrap Obj Colum*
	=MESSAGE(FALSE)	*Remove/clear message area*
	=RETURN()	

	SetObjColWidth	*Set width and rewrap Objective'*
arGetPageOrienta	=DIALOG.BOX(udGetPageOrientation)	*Get Portrait or Landscape*
	aPageWidth=IF(arGetPageOrientation=2,78.7,105)	*set width of page for portrait or*
	=IF(VALUE(GET.WORKSPACE(2))<4,IF(ALERT("Us	*Set Landscape/Portrait. User m*
	aWidthBetweenVisibleCells=0.717	*width Excel puts beween visible*
	aTotalWidth=0	
	=FOR.CELL("aTheCell","RC1:RC"&GET.DOCUMENT(	*Loop for all columns in the spre*
	aTotalWidth=aTotalWidth+IF(GET.CELL(16,aTheC	*calculate the total width NOTE:*

	=NEXT()	
	aTotalWidth=aTotalWidth-GET.CELL(16,!Col_Objec	*Remove the width of the Object*
	=COLUMN.WIDTH(MAX(20,aPageWidth-aTotalWidt	*set new width (with an enforced*
	=ROW.HEIGHT(,"R1:R"&GET.DOCUMENT(10),,3)	*rewrap all rows*
	=SET.NAME("aActCell",ACTIVE.CELL())	*save current active cell*
	aFirstPrintRow=ROW(!Print_Titles)+ROWS(!Print_T	*get row number of 1st row afte*
	=IF(aFirstPrintRow<=GET.DOCUMENT(10))	*any detail lines in document? (i*
	= SELECT("R"&aFirstPrintRow&"C1:R"&GET.DOCU	*select the region from after the*
	= ERROR(0)	*I will trap the SELECT.SPECIAL*
	= IF(NOT(ISERROR(IF(ROWS(SELECTION())=1,TRU	*from the current selection select*
	= ROW.HEIGHT(acMajorHeadingRowHeight)	*set the row height of the major*
	= END.IF()	
	= ERROR(1)	*Turn error checking back on*
	= SELECT(aActCell)	*restore active cell*
	=END.IF()	
	=SET.NAME("aActCell")	*remove temporary variables*
	=SET.NAME("aFirstPrintRow")	
	=SET.NAME("aTheCell")	
	=SET.NAME("aTotalWidth")	
	=SET.NAME("aWidthBetweenVisibleCells")	
	=SET.NAME("aPageWidth")	
	=SetPrint_Area()	*reset Print_Area incase user wa*
	=MESSAGE(FALSE)	*Remove/clear message area*
	=RETURN()	

	SetPrint_Area	
	=ECHO(FALSE)	*no screen update for speed*
	aFirstPrintRow=IF(ISERROR(GET.NAME("!Print_Titl	*get row number of 1st row afte*
	aLastPrintRow=GET.DOCUMENT(10)	*get row number of last print rov*
	aLastPrintRow=IF(aFirstPrintRow>aLastPrintRow,aF	*if nothing in document yet exce*
	=DEFINE.NAME("Print_Area","=R"&aFirstPrintRow&"	*set new print area*
	=SET.NAME("aFirstPrintRow")	*remove temporary variables*
	=SET.NAME("aLastPrintRow")	
	=RETURN()	

	GetDataFile	*Opens the specified objective da*
	=ECHO(FALSE)	*no screen update for speed*
	IPPDataFile="IPP "&CHOOSE(ud.ObjDB,"TMH","E	*construt name of data file*
	=IF(ISNA(MATCH(IPPDataFile,DOCUMENTS(),0)))	*file not yet open?*
	alPPDataFileSpec=GET.DOCUMENT(2,arThisMac	*get filespecs for data file (look i*
	= MESSAGE(1,"Loading IPP Objectives in "'&IPPD	
	= ERROR(0)	*I will trap the open error and HA*
	= IF(ISERROR(OPEN(alPPDataFileSpec,0,TRUE,,,,TR	*open file*
	= ALERT("Can't find file with IPP Objectives "'&	
	= HALT()	
	= END.IF()	

	= ERROR(1) = HIDE() = MESSAGE(FALSE) =SET.NAME("aIPPDataFileSpec") =END.IF() =RETURN()	*Turn error checking back on* *and hide it* *Remove/clear message area* *remove temporary variables*

	IPPAddIndent	*Indents all the cells in the curre*
	=ReIndent(8) =RETURN()	*Add indent(this number of spac*

	IPPRemoveIndent	*Indents all the cells in the curre*
	=ReIndent(0) =RETURN()	*Remove indent*

	ReIndent	*Adds or removes leading space*
	=ARGUMENT("aNumOfSpaces",1) =ECHO(FALSE) =FOR.CELL("aTheCell") aTheText=aTheCell = WHILE(LEFT(aTheText)=" ") aTheText=MID(aTheText,2,LEN(aTheText)) = NEXT() = FORMULA(REPT(" ",aNumOfSpaces)&aTheText, =NEXT() =SET.NAME("aTheCell") =SET.NAME("aTheText") =SET.NAME("aNumOfSpaces") =RETURN()	*Get number of spaces of indent* *no screen update for speed* *Loop for all columns in the spre* *Get contents of the cell* *remove leading spaces* *put the text back into the cell a* *remove temporary variables*

	IPPReformat	*Reformats the master IPP Objec*
	=ECHO(FALSE) =IF(TEXTREF("!R2C1")<>1) = IF(ALERT("Expecting the number '1' in the 1st =END.IF() =SET.NAME("aActCell",ACTIVE.CELL()) aFirstRow=IF(ISERROR(GET.NAME("!Print_Titles")) aLastRow=GET.DOCUMENT(10) =SetPrint_Area() =DEFINE.NAME("IPPCodeTable","=R"&aFirstRow&" =SELECT("R"&aFirstRow&"C1:R"&aLastRow&"C1") =ROW.HEIGHT(,,,3) =ALIGNMENT(,TRUE) =FORMULA(1) =DATA.SERIES(2,1)	*no screen update for speed* *check to make sure this is the r* *halt if user presses CANCEL bu* *save current active cell* *get row number of 1st row afte* *get row number of last row* *reset Print_Area incase user ad* *reset the area to search for obje* *select the region with the numb* *rewrap all rows in the region* *Turn on wrap for numbers so th* *start numbers at 1* *renumber Objectives*

	=ALIGNMENT(,TRUE)	*Turn on wrap for numbers so th*
	=SELECT("R"&aFirstRow&"C4:R"&aLastRow&"C4")	*select the region with the objec*
	=ALIGNMENT(,TRUE)	*Turn on wrap for objectives*
	=SELECT("R"&aFirstRow&"C3:R"&aLastRow&"C3")	*select the region with the Sub H*
	=ALIGNMENT(,FALSE)	*Turn off wrap*
	=SELECT.SPECIAL(2,2)	*select only cells with something*
	=APPLY.STYLE("Sub Heading")	*Format these cells*
	=SELECT("R"&aFirstRow&"C2:R"&aLastRow&"C2")	*select the region with the Main*
	=ALIGNMENT(,FALSE)	*Turn off wrap*
	=SELECT.SPECIAL(2,2)	*select only cells with something*
	=APPLY.STYLE("Main Heading")	*Format these cells*
	=ROW.HEIGHT(acMajorHeadingRowHeight)	*set the row height of the major*
	=FOR.CELL("aTheCell")	*Loop for all rows with Main Head*
	= SELECT(OFFSET(aTheCell,,-1))	*select the cell with the numbers*
	= ALIGNMENT(,FALSE)	*Turn off wrap for the numbers*
	=NEXT()	*loop back to get next cell*
	=SELECT(aActCell)	*restore active cell*
	=SET.NAME("aActCell")	*remove temporary variables*
	=SET.NAME("aTheCell")	
	=SET.NAME("aFirstRow")	
	=SET.NAME("aLastRow")	
	=RETURN()	

DIALOGS

type	x	y	wide	high	text	init/result	*names*
udGetIPPDialo							
			250	154	IPP Objectives		
1	178	12	64		OK		
2	178	43	64		Cancel		
7	77	123	171			1	*ud.IPPcode*
5	2	117	73	36	Objective Numbe		
13	103	102			Add Repeating	TRUE	*ud.AddRepeating*
14	12	6	118	85	Objectives		
11						2	*ud.ObjDB*
12					TMH		
12					EMH		
12	79	24			ELA		
12					DH		
3	26	63	93		View/Print		*ud.ViewPrintData*

udGetPageOri

			220	169			
5	9	27	205	19	How will the IPP t		
3	32	70	63	83	Portrait		
3	110	92	81	58	Landscape		

Appendix F

User Manual

This appendix contains the user manual created for use with the developed IPP computerized system (Wodelet, 1993).

IPP User Manual

Table of Contents

Section **Page**

Introduction 1

Target Audience 1

Hardware Requirements 1

Software Requirements 2

Installing the IPP Program 2

Starting a New IPP 3

Working With Objectives 4

Entering Objectives 4

Removing Objectives 5

Substituting Objectives 5

Changing the Wording of an Existing Objective 5

Inserting an Objective in Between Two Existing Objectives 6

Moving Through the File 6

Saving the IPP on Disk 6

Selecting a Format 8

Inserting Page Breaks 10

Printing the IPP 11

Customizing the IPP 12

Changing Column Headings 12

Changing Column Widths 12

Inserting New Columns 13

Changing How the Program Works 15
Sample Formats 16
Sample IPP Check List Format 17
Sample IPP Report Card Format 18
Sample IPP Achievement Format 19
Sample IPP Program Plan Format 20

List of Figures

Figure 1: IPP Diskette Icon 2
Figure 2: Diskette Contents 3
Figure 3: Student Name Dialog Box 3
Figure 4: Command Buttons 4
Figure 5: Insert Dialog Box 4
Figure 6: File Menu — Save As... 7
Figure 7: Save As Dialog Box 7
Figure 8: Page Orientation Dialog Box 8
Figure 9: Set Orientation Dialog Box 9
Figure 10: Page Setup Dialog Box 9
Figure 11: File Menu — Print Preview 10
Figure 12: File Menu — Print... 11
Figure 13: Changing Column Widths 13
Figure 14: Inserting a New Column 14
Figure 15: Program Source Code Layout 15

IPP User Manual

Introduction

The IPP Program will allow you to build an IPP for your student using objectives from the EMH, TMH, DH, or ELA list of objectives. Objectives from all four lists may be incorporated into one IPP if you wish. The Program also lets you choose from one of four formats to print your IPP. This IPP program may be used in conjunction with the Informed IPP program. It will replace part C (just put "See Attached" in part C of the Informed IPP file and attach a printout of one of the many formats you can produce with this file (see **Sample Formats** at end of document)).

Target Audience

This program is intended for use by any teacher involved with creating Individual Program Plans for students. It may be used for exceptional students as well as those in the regular classroom environment.

Hardware Requirements

This IPP program will run on either a Macintosh or IBM PC/clone capable of running Windows. Your system should have at least 2 megabytes of memory available but it may run with less. You may run it from a floppy drive but a hard drive is preferred. To install you will need at least 400 kilobytes of disk space available. The amount of disk space you actually need is dependent upon how may students you generate IPPs for — approximately 30 kilobytes / student.

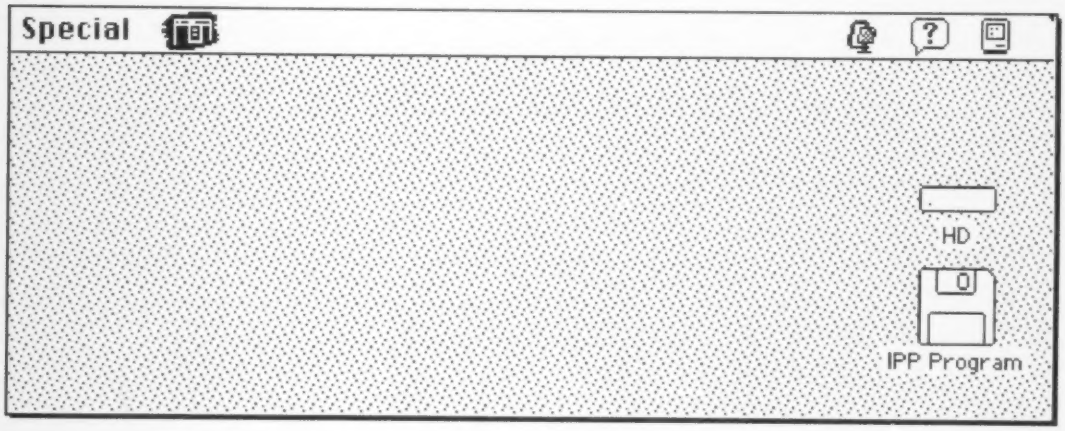
Software Requirements

To run the IPP program you must have Microsoft EXCEL version 3.0 or greater installed on your system. This is the only software you will require if you are running it from a Macintosh system. However, if you are running the IPP program from an IBM PC, or a related PC clone, you also require Microsoft Windows version 3.0 or greater.

Installing the IPP Program

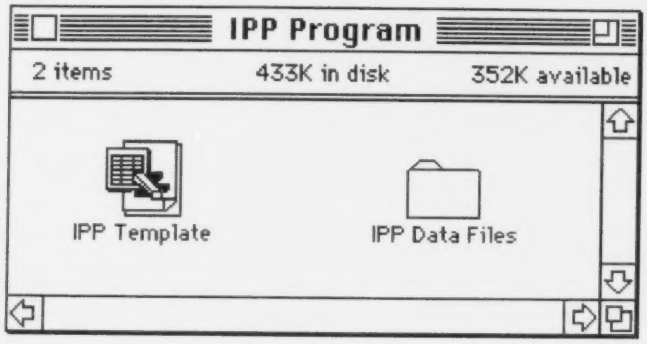
Insert your diskette containing the IPP Program into the computer. Double click on the disk icon on the right hand corner of the screen (See **Figure 1**).

Figure 1: IPP Diskette Icon



You will see a file called "IPP Template" and a folder called "IPP Data Files" (see **Figure 2**). Create a new folder on your hard disk (if you wish to put the files on your hard disk) and call it "IPP". Move this file and folder (on the floppy) into it. (For more information on moving files around between your floppy and your hard disk read your Macintosh or Windows manuals. The Macintosh also has an on-line tour tutorial that provides training on basic skills.)

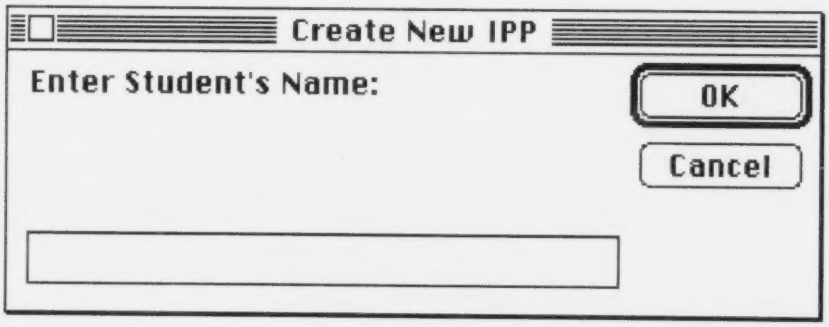
Figure 2: Diskette Contents



Starting a New IPP

Open the folder called "IPP" by double clicking on it (if you created this folder. If not, ignore that direction!). Open the file called "IPP Template" by double clicking on it. You are now inside the file and ready to start making the IPP. A dialog box will appear on the screen that says : "Enter Student's Name" (see **Figure 3**).

Figure 3: Student Name Dialog Box



Type in the name of the student you are creating an IPP for. When you have typed it in as you wish it to appear, click on "OK". The box will disappear and the name will appear at the top of the screen.

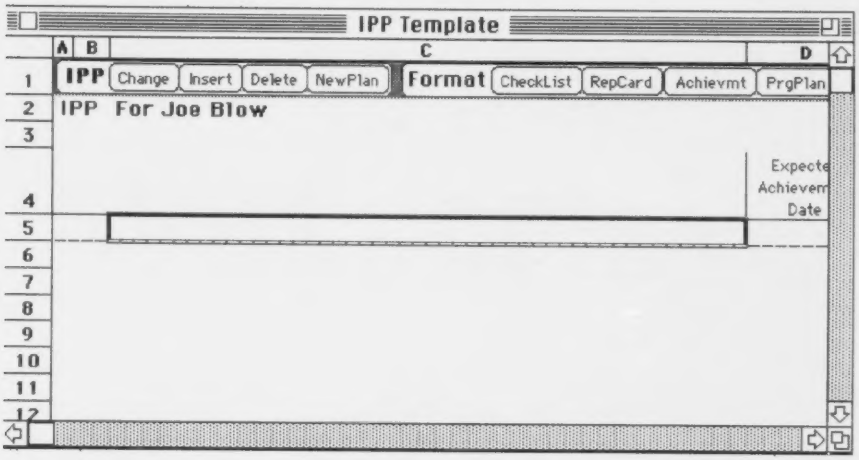
If you are already in an IPP file and you wish to begin work on another IPP (either after completing one, or while still working on another one) simply move the cursor to the button labeled "**New Plan**" and click on it. A new IPP template will appear on the screen for you to begin working in.

Working With Objectives

Entering Objectives

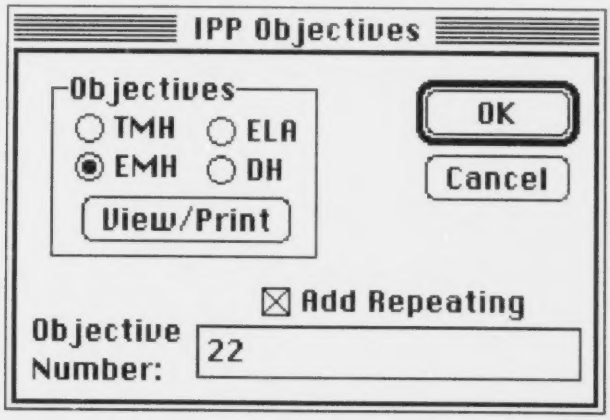
You are now ready to enter the specific headings and objectives for that student. To start adding objectives move the cursor to the buttons at the top of the file (see **Figure 4**) and click on the **Insert** button.

Figure 4: Command Buttons



A dialog box (see **Figure 5**) will appear on the screen asking you to choose which file you want to pick objectives from.

Figure 5: Insert Dialog Box



You may choose objectives from any or all of the four databases shown: **EMH**, **TMH**, **DH** and/or **ELA**. Click in the circle in front of the list you wish to select from. (If you want to use objectives from more than one, it is easy to switch back and forth by simply changing your selection here). If you want to view or print the objective database press the **View/Print** button (see **Figure 5**). Most people find it easier to add objectives by working from a printed copy of the objective file.

Now you are ready to type in the code number of the objective you wish to put in the IPP (the number in front of the objective or heading). Then click on "OK". With "Add Repeating" on (an "X"" in front of the "Add Repeating" box indicates it is on (see **Figure 5**)) the objective is entered into the IPP and the dialog box reappears for you to type the number of another objective in. With "Add Repeating" off (click on the box in front of "Add Repeating" to remove the "X" and turn it off - click again to turn it back on), the objective is entered and the dialog box does not reappear. If you wish to add more objectives, you will need to move the cursor to "Insert" again and click on it. You can have "Add Repeating" on or off as you wish.

Removing Objectives

Getting rid of an objective you have put in the IPP but now want to remove is easy. Move the cursor to the cell that the objective is in and click on that cell to select it (clicking on it makes it the active cell - you can tell which one is active by the dark line around it). Next, move the cursor up to the "Delete" button at the top (see **Figure 4**) and click on it. The objective is gone and the remaining objectives are automatically renumbered. If you want to continue to add more objectives to the end of the file, you need to select a cell in the row at the end of the file to be the active one (by clicking on it). Then click on the "Insert" button at the top of the screen. **Remember, by using the "Insert" button objectives are inserted in the line *above* the active cell. The "Delete" button deletes the objective line *containing* the active cell.**

Substituting Objectives

If you accidentally type in the wrong objective number or decide you want to change an objective once you have it in the IPP, it is easy to do. Move the cursor to the cell containing the objective you wish to change. Click on the mouse to select that cell. Now move the cursor up to the "Change" button (see **Figure 4**) and click on it. A dialog box will appear and you should type in the number of the new objective that you wish to replace the incorrect one with. Click on "OK and the new objective is substituted for the old one. **Any new objective added using the "Change" button is written *over top* of the objective in the selected cell, erasing the objective previously in that cell.**

Changing the Wording of an Existing Objective

Once you have put an objective in the IPP, it is easy to alter the wording of it if you wish to. Select the cell that the objective is in. The contents of the selected cell always appear

in a box at the top of the screen. Move the cursor up to that box and position the cursor (by clicking) in the place you want to make changes. You can now add in a word or phrase, delete a word (by using the delete key on the keyboard to delete characters), etc. Once the objective reads the way you want it to, press "Return". The changes are entered in the cell.

Inserting an Objective in Between Two Existing Objectives

Putting in an objective later is no problem. Move the cursor to the cell below where you want the new objective to go. Click on that cell to select it (to make it the active cell). Now add the objective in the usual manner (by clicking on the button "Insert" and typing in the number of the new objective in the dialog box). **Any new objective added using the "Insert" button is always positioned *above* the selected cell.**

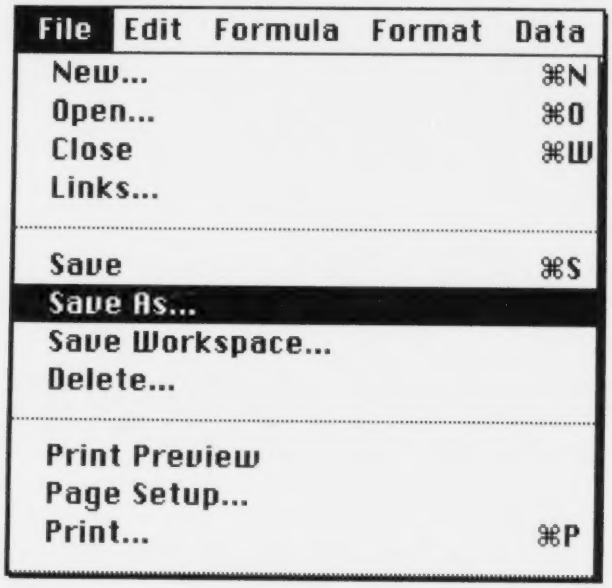
Moving Through the File

As with any file you can change your position on the screen by moving the cursor with the mouse and clicking. You may also use the arrow keys to move around. To move to a part of your file that isn't on the screen you will want to use the scroll bar (see **Figure 4**) at the right hand side of the file. Click above the box to move up in the file, below it to move down. To quickly move one way or the other, grab the box with the cursor (by clicking on it and holding the mouse button down) and drag it up or down. Use the scroll bar at the bottom of the screen in the same way to move sideways in the file.

Saving the IPP on Disk

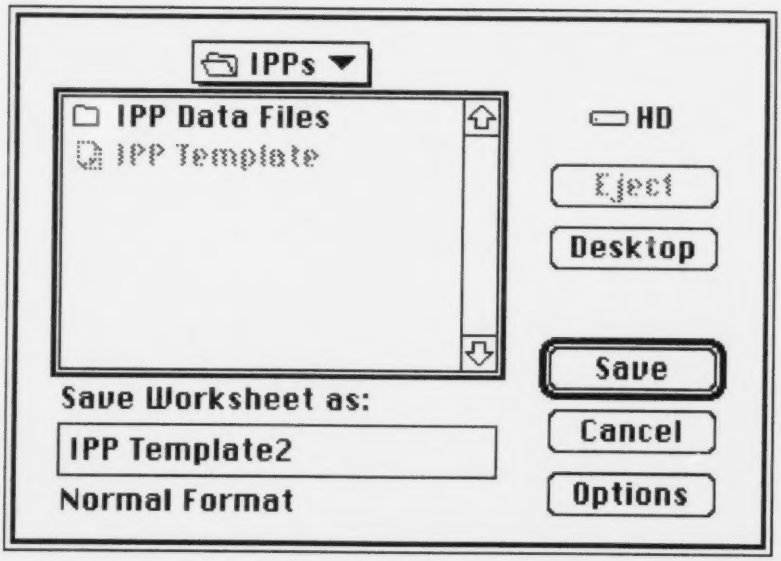
Move the cursor up to the menu bar at the very top of the screen. Click on "File" to view the menu underneath (hold the mouse down, or the menu will disappear). Still holding the mouse down, move the cursor down the menu until "Save As ..." is highlighted (see **Figure 6**).

Figure 6: File Menu — Save As...



Release the mouse. A dialog box will appear on the screen (see **Figure 7**). You are asked to type in a name for your file. It is recommended that you use the student's name as the name of the file, with the date incorporated as well: e.g.. Smith/92-93. After you have typed in the name you want, click on "Save". Your file will be saved to the disk.

Figure 7: Save As Dialog Box



As usual, **it is a good idea to save frequently as you work** to minimize work lost in the event of a power failure. If you forget to save your file to disk before you "Quit", you will be asked to do so when you try to quit. (To save once you have named the file, just use "Save" under the "File" menu (or Command "S")). It is also a good idea to make a backup copy of your files, especially if you are storing them on a floppy disk. Occasionally disks do go "bad".

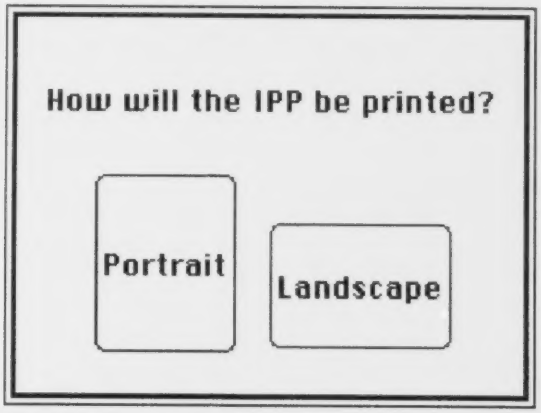
Note: Your student files should be saved beside the original "IPP Template" file. **DO NOT** put them in a separate folder.

Selecting a Format

There are presently four formats that you can print your IPP in: **Check List**, **Report Card**, **Achievement** and **Program Plan**. To view each format in order to decide which one you want, move the cursor to the "Format" button (see **Figure 4**) that you wish to view and click on it (samples of each format appear at the end of this document). When you click on one of the format buttons, the screen changes to show you the format that will print when that one is selected. You can, of course, print the same IPP in all four formats for different purposes.

When you choose one of the four format buttons you are also asked: "How will the IPP be printed?" (see **Figure 8**). The choice is "Portrait" or "Landscape" (vertically or horizontally).

Figure 8: Page Orientation Dialog Box



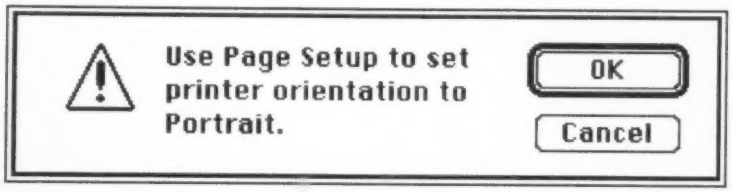
Because of the size of columns, you will probably only want to select "Landscape" for the "Program Plan" format. For

"Check List"	you select	"Portrait" or "Landscape"
"Achievement"	you select	"Portrait" or "Landscape"
"Report Card"	you select	"Portrait" or "Landscape"
"Program Plan"	you select	"Landscape"

Click on the appropriate box to select the correct page orientation you want to use.

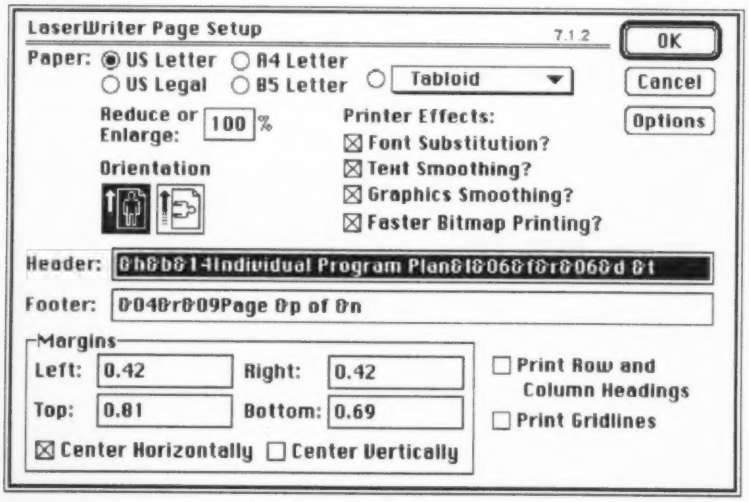
If you are using Excel version 3.0 or less another dialog box appears (see **Figure 9**) telling you to "Use Page Setup to set printer orientation to Portrait or Landscape". Click on "OK". If you are using Excel version 4.0 or greater the program is able to do this automatically for you so the dialog box shown in **Figure 9** or **Figure 10** will not appear.

Figure 9: Set Orientation Dialog Box



Another dialog box then appears (see **Figure 10**) in which you can change options for page setup. Locate "Orientation" and click on the correct one (if the correct one is not already highlighted). Click on "OK" again to return to your file. (To actually print the IPP read the instructions further on for printing. You are, however, ready to print now).

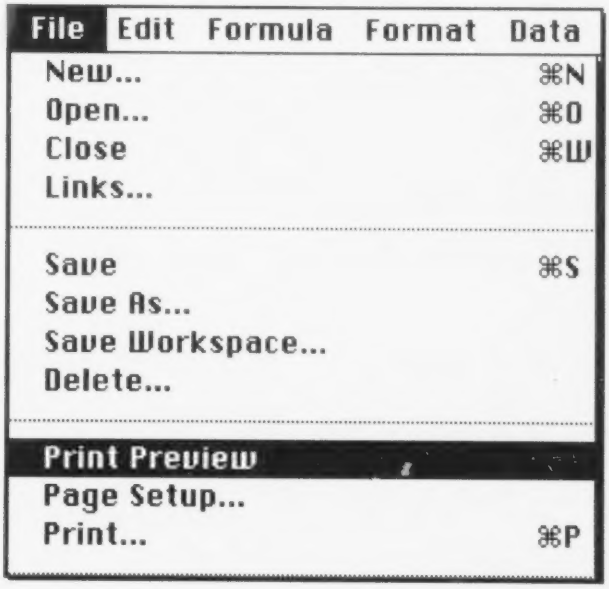
Figure 10: Page Setup Dialog Box



Inserting Page Breaks

If your IPP is more than one page long you may wish to select where one page ends and the next begins by inserting a page break. If you do not do this, a new page will begin whenever the previous one is full. In order to see where new pages will begin, it is necessary to do a "Print Preview". Go up to the menu bar at the very top of the screen and click on "File" (continue to hold the mouse down). Still holding the mouse down, drag the cursor down to "Print Preview" (see **Figure 11**) until it is highlighted and then let go of the mouse to select it. You can now view your document as it will appear when it is printed. If you decide you want to change where a new page will begin, do it as follows.

Figure 11: File Menu — Print Preview



Get out of "Print Preview" and back into the file by clicking on "Close". Move the cursor over to the **very left hand edge of the screen (on the row number), just below** where you want the break to be. Click the mouse to select that row (the whole row below where you want the break to be). (If you select only one cell in that row, rather than the whole row, you may insert vertical as well as horizontal breaks, which you do not want. That is why it is important to make sure the whole row is selected.) Now move the cursor up to the menu bar again and click on "Options". Hold the mouse down to view the menu. Still holding the mouse down, drag the cursor down the menu to "Insert Page Break". When it is highlighted, let go of the mouse to select it and a page break will be inserted above where the cursor is positioned in the document. To check it out, you can do another "Print Preview". Note: Every time you change formats you may wish to

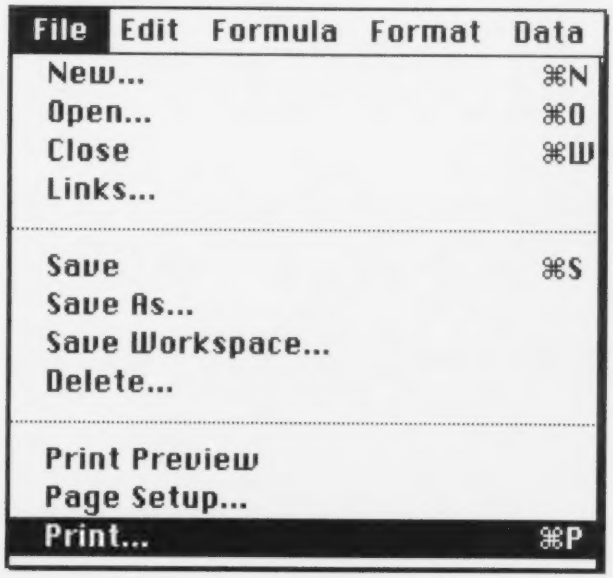
check and see where your page breaks are (because the column width changes, the page breaks may no longer be appropriate). To remove the page break, position the cursor below the break and under the "Options" menu select "Remove Page Break".

NOTE: If you go to "Print Preview" and what you see there does not match what is in your file (i.e. some stuff is missing!), don't panic. Go back to your file and format it again by clicking on the format that you wanted it printed in (do this even if it appears in that format on the screen). When you go back to "Print Preview" the whole file should now appear.

Printing the IPP

Again, move the cursor up to the menu bar at the very top of the screen. Click on "File" and hold the mouse down while you move down the menu to "Print..." (see **Figure 12**). Release the mouse to select "Print..." and make the selections you wish to in the dialog box. If you leave it as it is, you will print one copy of the whole document. When you are ready to print it, click on "Print".

Figure 12: File Menu — Print...



Customizing the IPP

One of the strongest features of this IPP system is the ability to customize it to meet your own specific needs. In previous sections you learned how to insert, delete and change objectives to meet the specific needs of the student. In this section you will learn how to change/customize how the information is presented to meet the specific needs of the teacher or the school system.

Changing Column Headings

To change a column heading it is a simple matter of selecting an existing column heading with your mouse and typing in your new column heading. If you wish this to be done for all new IPPs you create, you must change the **IPP Template**[1] file. The Excel manual describes in detail how to open and change template files. Briefly, to open a template file you must use the "Open..." option under the "File" menu. If you hold down the shift key when pressing the "Open" button using the mouse the actual template file will be opened, rather than a copy of it. Once the template file is opened you can modify it in the same way as you would modify any other Excel worksheet.

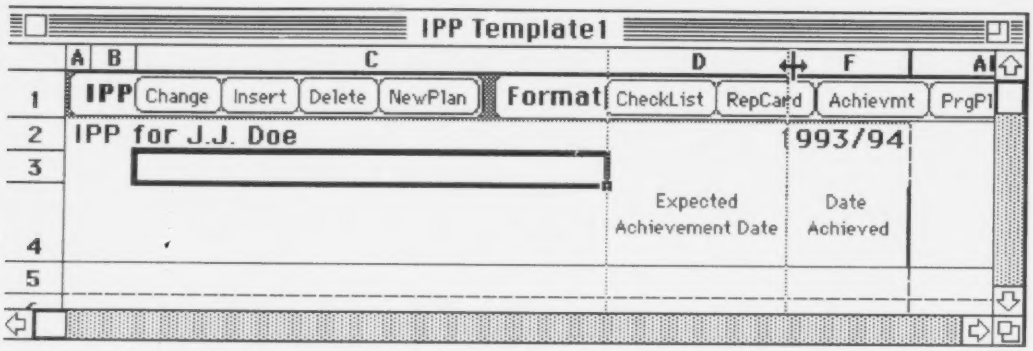
Changing Column Widths

The width of any column of the IPP can be easily changed. Just move the mouse pointer into the column heading and close to the right hand side of the column (close to the column dividing line). The mouse pointer will change from a ✚ to a ✣ indicating you can change the column width (see **Figure 13**). Simply hold the mouse button down and move the right hand dividing line of the column to make the column bigger or smaller. You can change the column width of a number of columns at once to be all the same size (such as in the **Check List** format) by selecting all the columns you want to change (click and drag over all the column headings to select) and then adjusting the column width of one of the columns. When the width of one of the selected columns is adjusted, all the other selected columns will also be adjusted to be of exactly the same width.

---

[1] The **IPP Template** file is a special type of file in Excel. When a template file is opened a copy of the file is automatically made and opened instead of the original file, thereby protecting the actual template from inadvertent modification. Because of this unique feature of template files, a special procedure must be used if you need to open and modify the actual template file. The Excel manual describes template files in more detail.

Figure 13: Changing Column Widths



Remember, to change the width of a column you must pick the *right hand side* of the column to move. The above figure shows the width of column "D" being changed.

After changing column width you **must** reformat your IPP. This will cause the IPP program to automatically re-adjust the column width of the objective column so that the IPP will again fit on a single page.

The only column width that the program readjusts automatically when changing formats is the column containing the objectives. This column width is set automatically to the largest size possible depending on the width of the other columns and your page orientation. For example when you switch from **portrait** to **landscape** page orientation, this column becomes wider to accommodate the wider page size in landscape mode. However, if you increase the column width of any of the other columns and reformat (by pressing one of the format buttons along the top of the window (see **Figure 13**)) the width of the objective column will shrink to accommodate the now larger column on the page.

There is a minimum column width enforced for the objective column to prevent it from becoming too small. Therefore, if you make the other columns too big the IPP may become too wide to fit on a single page. You will either have to reduce the width of some of the other columns, change page orientation from portrait to landscape, and/or use "Page Setup..." (under the "File" menu) to shrink the printed page.

If you wish to change the column width of some of the columns for all new IPPs you create you must change the **IPP Template** file. The Excel manual describes how to open and change template files. This process is also discussed briefly in the section **Changing Column Headings**.

Inserting New Columns

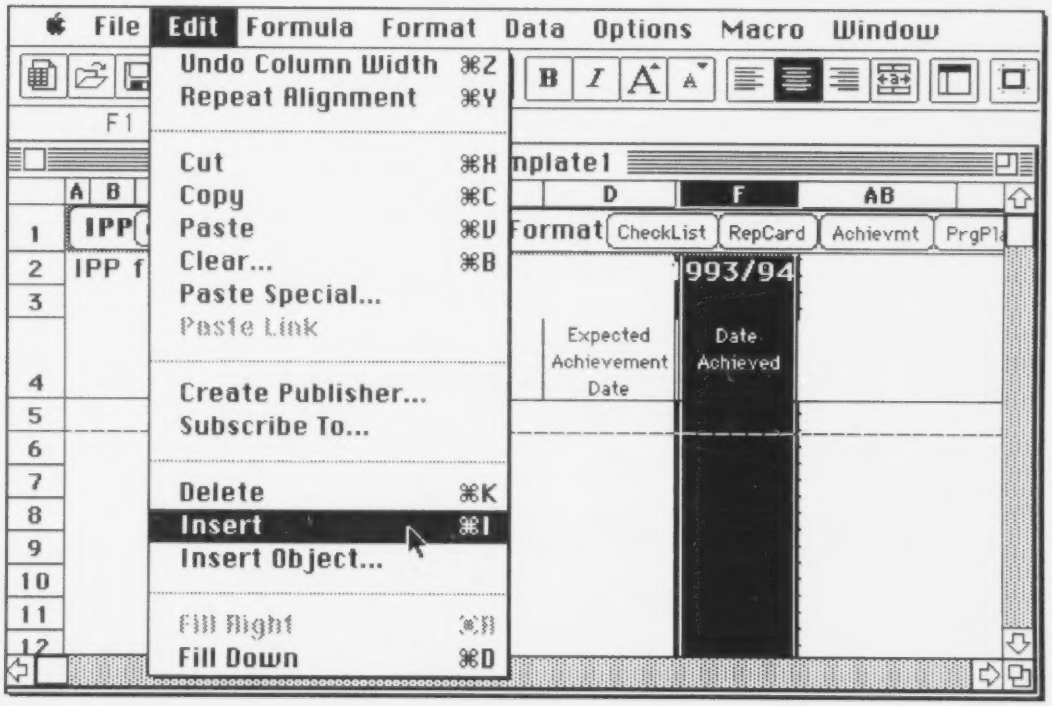
To add a new column to an IPP you first need to select an existing column which is directly **to the right of** the place you want the new column to be. Once this column is selected, use the **Insert** option under the **Edit** menu to insert the new column (see **Figure 14**). (A short cut is to hold the option key down while you select the column to the right

of where you want the new column to be — this will cause a new column to automatically be inserted.)

After inserting a new column you **must** reformat your IPP. This will cause the IPP program to automatically re-adjust the column width of the objective column so that the IPP will again fit on a single page. If you add too many columns the program may not be able to make the objective columns small enough (a minimum size is enforced) and still fit the IPP on the width of a single page. If this happens you need to either adjust the widths of the existing columns, delete some columns, change from portrait to landscape orientation, and/or shrink the printed page size by using "Page Setup..." (under the "File" menu).

If you wish to have these new columns show up for all new IPPs you create you must change the **IPP Template** file. The Excel manual describes how to open and change template files. This process is also discussed briefly in the section **Changing Column Headings**.

Figure 14: Inserting a New Column



Remember, to add a new column you must select the column *to the right of* the new column you want to add.

Changing How the Program Works

Full customizability is provided through access to the Excel Macro programming language. The program is completely written in this language and you are provided with the complete documented source code to allow you to customize the program specifically for your needs. For example, by modifying the program code you will be able to add additional formats and format buttons (as displayed along the top of the IPP window). The Excel **Function Reference Manual** that comes with the Excel program provides a complete description of the macro language. However, depending on the expertise of the user and the extensiveness of the modifications it could take anywhere from days to weeks to become fully fluent in this language. There are a number of other books and video tapes (Mac Academy) available to aid in learning the language.

The Excel macro source code is normally hidden from view to prevent unnecessary screen clutter and prevent accidental modification. To access the source code file ("**IPP Macros**") you can double click on this file directly (it's in the "**IPP Data Files**" folder). Alternatively, if you are currently working on an IPP, you can use the **Unhide** option under the **Window** menu to unhide the source code file. The source code is fully formatted and documented to help you understand what each line of source code does (see **Figure 15**). The first column is used to keep (and self-document) reference names which are used for the target of the GOTO function or to reference a return value from a function call which is needed elsewhere in the program. Even though the names are contained in the first column, they actually reference the adjacent cell in the second column. The second column contains the actual macro source code. The comments and documentation for each line of source code is contained in the third column.

Figure 15: Program Source Code Layout

names	*commands*	*comments*
	auto_open	
arThisMacroFileN	=GET.CELL(32,A4) =FULL(TRUE) =WINDOW.SIZE(560,420,arThisMacroFileName) =RETURN()	*get name of this document* *reverse size change by auto_clos*
	IPPNewPlan	*Creates a NEW IPP Plan/worksh*
arNewStudentNam	=ECHO(FALSE) =INPUT("Enter Student's Name:",2,"Create New IP =IF(arNewStudentName,,HALT()) =CreateNewPlan(TRUE) =RETURN()	*no screen update for speed* *get student's name* *quit if Cancel button pressed* *Create the new plan*

Sample Formats

Sample IPP Check List Format

IPP for <student name goes here>	1992/93														
	Expected Achievement Date														
Communication															
1. Listens for the purpose of recalling details															
2. Listens attentively for a specified period of time and appropriately answers questions when called upon - 20 minutes															
3. Sequences 3 or 4 concepts/pictures/ideas/instructions															
4. Draws pictures and writes about a familiar activity in sequential order															
5. Demonstrates ability to listen in a distracting situation - 10 minutes															
Syntax/Language Concepts - Comprehends and Uses:															
1. Pronouns - me, mine, my, I you, he, she, they, his, her, their, myself, her, him, herself, himself															
2. Copula verbs - is, are, am															
3. Prepositions - up, down, top of, bottom of, in, out, on, under, above, below, beside, behind, between, in front, by															
4. Conjunctions (and, but, because, or)															
Computation															
1. When shown two sets (objects and number symbols), indicates which set contains "more" and which set contains "less"															
2. When shown two numerals indicates which is "more" and which is "less"															
3. Demonstrates with concrete materials that addition is a joining proces, using no more than 5 objects															
Science															
Matter															
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide															
Senses															
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide															

Sample IPP Report Card Format

IPP for <student name goes here>				1992/93
	Expected Achievement Date	Term 1	Term 2	Term 3
Communication				
1. Listens for the purpose of recalling details				
2. Listens attentively for a specified period of time and appropriately answers questions when called upon - 20 minutes				
3. Sequences 3 or 4 concepts/pictures/ideas/instructions				
4. Draws pictures and writes about a familiar activity in sequential order				
5. Demonstrates ability to listen in a distracting situation - 10 minutes				
Syntax/Language Concepts - Comprehends and Uses:				
1. Pronouns - me, mine, my, I you, he, she, they, his, her, their, myself, her, him, herself, himself				
2. Copula verbs - is, are, am				
3. Prepositions - up, down, top of, bottom of, in, out, on, under, above, below, beside, behind, between, in front, by				
4. Conjunctions (and, but, because, or)				
Computation				
1. When shown two sets (objects and number symbols), indicates which set contains "more" and which set contains "less"				
2. When shown two numerals indicates which is "more" and which is "less"				
3. Demonstrates with concrete materials that addition is a joining proces, using no more than 5 objects				
Science				
Matter				
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide				
Senses				
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide				
Phases of Matter				
1. Reviews classifying hot and cold things				
2. Identifies solids/liquids/gases				
3. Observes that matter can change from one state to another				
4. Participates and observes in class discussions and experiments				

Sample IPP Achievement Format

IPP for <student name goes here>		1992/93
	Expected Achievement Date	Date Achieved
Communication		
1. Listens for the purpose of recalling details		
2. Listens attentively for a specified period of time and appropriately answers questions when called upon - 20 minutes		
3. Sequences 3 or 4 concepts/pictures/ideas/instructions		
4. Draws pictures and writes about a familiar activity in sequential order		
5. Demonstrates ability to listen in a distracting situation - 10 minutes		
Syntax/Language Concepts - Comprehends and Uses:		
1. Pronouns - me, mine, my, I you, he, she, they, his, her, their, myself, her, him, herself, himself		
2. Copula verbs - is, are, am		
3. Prepositions - up, down, top of, bottom of, in, out, on, under, above, below, beside, behind, between, in front, by		
4. Conjunctions (and, but, because, or)		
Computation		
1. When shown two sets (objects and number symbols), indicates which set contains "more" and which set contains "less"		
2. When shown two numerals indicates which is "more" and which is "less"		
3. Demonstrates with concrete materials that addition is a joining proces, using no more than 5 objects		
Science		
Matter		
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide		
Senses		
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide		
Phases of Matter		
1. Reviews classifying hot and cold things		
2. Identifies solids/liquids/gases		
3. Observes that matter can change from one state to another		
4. Participates and observes in class discussions and experiments		
Air		
1. Understands that air moves		
2. Understands that moving air can "push" things		

Sample IPP Program Plan Format

IPP for <student name goes here>		1992/93
	Intervention Strategies	Evaluation Criteria
Communication		
1. Listens for the purpose of recalling details		
2. Listens attentively for a specified period of time and appropriately answers questions when called upon - 20 minutes		
3. Sequences 3 or 4 concepts/pictures/ideas/instructions		
4. Draws pictures and writes about a familiar activity in sequential order		
5. Demonstrates ability to listen in a distracting situation - 10 minutes		
Syntax/Language Concepts - Comprehends and Uses:		
1. Pronouns - me, mine, my, I you, he, she, they, his, her, their, myself, her, him, herself, himself		
2. Copula verbs - is, are, am		
3. Prepositions - up, down, top of, bottom of, in, out, on, under, above, below, beside, behind, between, in front, by		
4. Conjunctions (and, but, because, or)		
Computation		
1. When shown two sets (objects and number symbols), indicates which set contains "more" and which set contains "less"		
2. When shown two numerals indicates which is "more" and which is "less"		
3. Demonstrates with concrete materials that addition is a joining proces, using no more than 5 objects		
Science		
Matter		
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide		
Senses		
1. Follows grade 3 objectives with modified assignments and assistance from teacher aide		
Phases of Matter		
1. Reviews classifying hot and cold things		
2. Identifies solids/liquids/gases		

Appendix G

IEPWorks: Sample IPP

This appendix contains a sample IPP from the IEPWorks computerized system for the management of IPPs. This IPP was generated using the actual databases provided with the system. However, all names and other personal information it contains are factitious.

Edmonton Public Schools
Edmonton, Alberta

1990-91 IEP for Norton

Attendance Center: _____
IEP Coordinator: _____

English as a Second Language Outcome 1: Students have the ability and desire to read, listen, view, and discuss for the variety of purposes.

ESL1.1 Norton will use strategies for establishing meaning when reading, listening, viewing, and discussing.
Grade 3 students
- label and categorize concrete objects

TARGET DATE PROGRESS REVIEW/COMMENTS
November-working on vocabulary development with respect to the home, including names and function of rooms, furniture, and appliances
March-worked on vocabulary relating to meals and body parts
-Norton experiences continued difficulty with labeling colors
June-have worked on labeling and categorizing foods, people/occupations and emotions
-Norton will need review of all of these next school year

English as a Second Language Outcome 4: Students gain fluency and control of both spoken and written language in a variety of situations.

ESL4.2 Norton will recognize and use accepted organizational patterns and conventions.
Grade 3 Students
- Show increasing competency in applying appropriate language structures to their own speaking and writing

TARGET DATE PROGRESS REVIEW/COMMENTS
November-working on using proper pronouns with respect to gender.
March-continue working on he/she
June -review he/she in September
-have worked on proper use of I, me mine, my. Review in September

Mathematics Outcome 1: Students demonstrate an understanding of mathematical knowledge.

M1.1
Norton demonstrates an understanding of whole number numeration and simple fractions.
Grade 3 Students
- order whole number (0 - 1000)

TARGET DATE	PROGRESS REVIEW/COMMENTS
November	-working on numbers for 1 to 5. Continue working on.
March	- rote counts to 10 - recognizes 1 and 2 sometimes - matches like numerals
June	- recognizes 1,2, and 3 -rote counts to 20 -continue working on this objective in 1992/1992 school year

Mathematics Outcome 3: Students demonstrate an awareness of the patterns, relationships, and broad principles of mathematics.

M3.1
Norton can describe, extend, and create a variety of patterns and
relationships.
Students
- identify numbers and shape patterns
TARGET DATE PROGRESS REVIEW & DATE/COMMENTS
November-knows numbers 1 and 2. Continue working on.
March-identifies, extends and creates simple patterns with objects
-continue working on colors and shapes patterns
June-continue working on this objective in 1991/1992 school year

Health Outcome 3: Students demonstrate emotional growth in developing personal and social relationships.

H3.6
Norton will demonstrate appropriate social skills in the following areas of the Walker skills program.

AREA 3 - Getting Along Skills

1. Norton demonstrates using polite words.

2. Norton demonstrates sharing

3. Norton demonstrates following rules.

4. Norton demonstrates assisting others.

5. Norton demonstrates touching the right way.

TARGET DATE PROGRESS REVIEW/COMMENTS

November-working on #1 and #5. Continue to work on these.

March-1. mastery
2. sharing is not consistent
3. follows classroom rules, but does not always follow rules outs
4. very conscientious and helpful
5. not always successful. We are presently working on the idea that even nice touches must be done gently and for short times (i.e.. hugs must not be too long or too hard)

June-1 & 3 mastery
2. continue working on in 1991/1992 school year
4. needs reminders to ask person before helping (must not assume someone wants help)
5. continue working on in 1991/1992 school year

AREA 5 - Coping Skills

1. Norton demonstrates what to do when someone says no.

2. Norton demonstrates what to do when you express anger.

3. Norton demonstrates what to do when someone teases.

4. Norton demonstrates what to do when someone tries to hurt you.

5. Norton demonstrates what to do when someone asks you to do something you can't do.

6. Norton demonstrates what to do when things don't go right.

TARGET DATE PROGRESS REVIEW/COMMENTS

November-working on #1. Continue working on as generalization beyond the classroom is rare.

March-1. can verbalize what to do but does not respond consistently

2. demonstrates- telling the person why you are angry
- asking them to stop
does not always do these spontaneously

3 & 4. demonstrates but does not always do spontaneously

5 & 6. not yet covered

June-continue working on all of these skills in 1991/1992 school year
-focus on helping Norton generalize these skills to the conflicts he
experiences outside

Language Arts Outcome 7:
Norton can recognize that writing can be used to record and share experiences.
Grade 3 Students
- identifies lower case letters out of sequence
- identifies upper case letters out of sequence
- matches upper and lower case letters

TARGET DATE PROGRESS REVIEW/COMMENTS

November -inconsistent on the letters of his name. Continue working on.

March-can spell his name by rote memory
-does not recognize upper or lower case letters by name
-shows no receptive recognition either
-is beginning to match upper to upper and lower to lower case

June-spells name by rote memory
-matches upper to upper and lower to lower with help
-beginning to identify letters receptively, but not yet expressively
-continue working on this objective in 1991/1992 school year

I.E.P. Development Participant Signatures:

Parents(s)_____ Date_____

IEP Coordinator_____ Date_____

Principal_____ Date_____

Student_____ Date_____

PARENTAL CONSENT

I am familiar with the goals and objectives included in the Individual Education Plan.

Parent/Lawful Custodian_____ Date_____

Appendix H

IEP Database: Sample IPP

This appendix contains a sample IPP from the IEP Database computerized system for the management of IPPs. This IPP was generated using the actual databases provided with the system. However, all names and other personal information it contains are factitious.

Special Education Placement and Annual Review
Spring Review 1991-92

Student: Norton Doe **School:** Central Office
Birthdate: Sept. 28, 1963 (16 yrs)
Teachers: T. Black, D. Brown, A. Yellow, A. Green, E. Violet, S. Browne, V. Blue
Parents(Guardians): Murgatroid & Brunhilda Doe
Address: RR#1 Pleasantville Lunenburg County, N.S. B0J 2C0
Telephone: 543-2583(H) 543-2503(W)
Home School: Hebbville Junior High **Transportation:** Bus #41

A. **Present Placement:** High Needs, BHS
Present Mainstreaming: Grade 8 Physical Education and PDR

B. **Background information on current functioning level and achievement.**

Functions independently in self help skills.
Social skills are appropriate for a young teenager. (Weird taste in music)
Mathematics skills are at Grade 6 level. (Brigance Red)
Reading comprehension at Grade 3 level. (Brigance Red)
Oral communication skills appropriate for Grade 4 level.

Areas of Strength.

none

Areas of Weakness.

none

C. Special Services:

	Recommended	Provided	Frequency	By whom
Psychoeducational Assessment	√	√	1/2yr	G. Brewer
Speech & Language Assessment	√	√	1/2yr	D. Ramey
Speech & Language Therapy	√	√	1/2wk	K. Moore
Occupational Therapy	√	√	1/2yr	SSRH
Physiotherapy	√	√	1/2yr	SSRH
Services, Visually Impaired	√	√	1/2yr	APSEA
Services, Hearing Impaired	√	√	1/2yr	APSEA
Services, Psychological	√	√	1/2yr	SSMH
Behavioural Consultation	√	√	1/2yr	SSMH

D. Long Term Goal Statement: Independent living & employment

Future Programming Needs: High Needs/ Work Experience

E. Recommended Placement: High Needs, Work Experience Class.
Proposed Mainstreaming: Grade 8 Physical Education and PDR
Effective Placement Date: September, 1992
Approval and Signatures:

Date: _____
Parent: _____
Principal: _____
Teachers: _____

F. Follow-up Date: June, 1993
Person responsible for arranging follow-up: J. Orange
Person responsible for monitoring: A. Black
Parental liaison: C. Helper

ACADEMIC SKILLS

GOAL	LONG TERM OBJECTIVES	SHORT TERM OBJECTIVES

2. Developing Reading Skills.

    2.1 Develop Orientation to Reading.
        e. Imitate language patterns (i.e., chant, sing, dance, improvise).
        f. Mimic reading process.
        g. Demonstrate voice word match.

    2.2 Develop Visual Analysis Skills.
        d. Show orientation to page, directionality.

3. Developing Writing Skills.

    3.3 Develop/Increase Spelling Skills in Context.
        e. Spell at a level to meet functional needs.
        f. Spell at a specific grade level.

    3.4 Develop Written Language Usage.
        b. Demonstrate appropriate use of capital letters.
        h. Demonstrate use of abbreviations.
        i. Demonstrate use of possessives.
        j. Demonstrate recognition and use of suffixes.

ACADEMIC SKILLS

PROGRAMS/ STRATEGIES	EVALUATION CRITERIA	STAFF RESPONSIBLE
2. Developing Reading Skills.		
2.1 **Develop Orientation to Reading.** All Aboard for Readiness Buzzing into Readiness Developing the Early Learner-Level 1 Readiness Joy Reading-PrePrimer B Reading Primer Reading Readiness	(this column to be filled out by the teacher)	(this column to be filled out by the teacher)
2.2 **Develop Visual Analysis Skills.** All Aboard for Readiness Buzzing into Readiness Developing Visual Skills Gr. K-2 Readiness Joy		
3. Developing Writing Skills.		
3.3 **Develop/Increase Spelling Skills in Context.** Basic Goals in Spelling Canadian Spelling Program - Ves Thomas I Can Spell Megawords, Books 1-8 Morphographic Spelling		
3.4 **Develop Written Language Usage.** correct language usage before written requirement mnemonic devices:" comma, quotation mark, capital " punctuation in student writing rule: verb end in 's ', noun doesn't syllables: clap beats, hand under chin		

DEVELOPMENTAL SKILLS

GOAL	LONG TERM OBJECTIVES	SHORT TERM OBJECTIVES

1. Developing Perceptual Motor Skills.

    1.1 Develop Gross Motor Skills (Locomotion).
        a. Move with wheelchair/walker.
        e. Walk up and down stairs:
            1. same foot leading.
            2. alternating feet.

    1.2 Develop Gross Motor Skills (Fitness and Ability).
        e. Roll forward.
        j. Create movement in dance.

    1.3 Develop Gross Motor Skills (Ball).
        h. Reach and grasp ball.

2. Developing Sensory Integration Skills.

    2.1 Develop Visual Integration.
        a. Demonstrate ability to attend to visual presentations.
        j. Identify between like/unlike objects.

DEVELOPMENTAL SKILLS

PROGRAMS/ STRATEGIES	EVALUATION CRITERIA	STAFF RESPONSIBLE

1. **Developing Perceptual Motor Skills.**

1.1 **Develop Gross Motor Skills (Locomotion).** (these columns to be filled out by the teacher)
use of air mattresses
use of games, dance and sports as incentive
vary the length of sessions
vary the speed and tempo

1.2 **Develop Gross Motor Skills (Fitness and Agility).**
Brigance Inventory of Early Development
Brigance Readiness: Strategies and Practices
weights

1.5 **Develop Gross Motor Skills (Equipment).**
adaptive equipment: stairs, tires, hoops, wands, cargo nets, batting tees, stationary bicycle, horizontal ladders, air mattresses, mats, oversized ball, trampoline, plastic balls for pucks

3. **Developing Communication Skills.**

3.1 **Develop Listening, Social Communication and Vocabulary Skills.**
All Aboard for Readiness Skills

Appendix I

IPP Information Form

This appendix contains a sample of the IPP Information form designed to be used in conjunction with the developed IPP system (Wodelet, 1993). The computerized system developed for this research project was not intended to be a complete IPP system. Although it is fully functional and can be used stand-alone, it should in the context of a larger framework which collects and maintains other needed information. At the XXX School District it complements this existing form which collects this additional needed information. This IPP information form is currently in the process of being computerized using a commercial forms package. When the IPP system is used with this form, section C is not filled out. Rather, the phrase "See Attached" is entered in this box and a printout of the student's IPP from the computerized system is attached to the form.

Individualized Program Plan

Name _____ Age _____

DOB _____

School _____ Grade/ Course _____

Parent/ Guardian _____ Phone _____

Student Support Team Members	Position

Type of Individualized Program or Intervention: _____

IPP reviewed with parent/guardian ☐ Date: _____ Signature of Parent/Guardian _____

A: Assessment of current performance level. Data gathered through:

Observation ☐ Interview ☐ Assigned tasks ☐ Other ☐ _____

Standardized Tests a) _____ b) _____ c) _____

Additional pertinent information: _____

General modifications required in curriculum delivery:

B. Data Interpretation: Instructional Focus: _____

Skills and Strategies	Study/Learning Behaviors
Strengths	Strengths
Areas for development	Areas for development

C. Identified Learning Outcomes	Intervention Strategies	Evaluation Criteria	Responsibility	

Projected Timeline: _____ to _____

D. Record of progress toward intended

Review schedule: Every _____ weeks

Outcome #	Date	Observable results, changes, growth	What is working?	Signature

Intervention Period: _____ to _____

E. Evaluation and Recommendations

Learning Outcomes Achieved (Celebrations)	Intervention Strategies that Worked
Goals/Focus for continued support	**Support services required** (personnel, materials, classroom & equipment modifications)

Final review meeting attended by: _____ _____

_____ _____

Summary prepared by: _____ **Date:** _____

Curriculum Vita

Curriculum Vita

of

David L. Wodelet

Full Name:	David Lyle Wodelet
Address:	33 Chelsea Way Sherwood Park, Alberta Canada, T8H 1E6
Telephone:	Office: (403)464-8078 Home: (403)467-9645
Date of Birth:	July 23, 1955
Nationality:	Canadian
Family Status:	Married, 2 children (born 1985 and 1987)

EDUCATION

Institution	Dates	Course / Specialization
University of Alberta Edmonton, Alberta	1992-93	M.Ed. Instructional Technology (Thesis route)
University of Alberta Edmonton, Alberta	1981-82	M.Sc. Computing Science (Partially completed)
University of Alberta Edmonton, Alberta	1977-79	B.Ed. Secondary.Science/Computing (First Class Standing)
University of Alberta Edmonton, Alberta	1973-77	B.Sc. Biology/Genetics (Honors)
Penticton Secondary Penticton, B.C.	1970-73	Academic Program

DECUS Seminars, Anaheim	May	1992	Network Systems Management
Learning Tree International,	Aug.	1991	Design and Development of Window Systems
DECUS Seminars, New Orleans	May	1990	X Window Systems and Servers
Learning Tree International,	Feb.	1990	ISDN (Integrated System Digital Network)
DECUS Seminars, Atlanta	May	1989	X Window Development
DECUS Seminars, Cincinnati	May	1988	MAC/VAX Connectivity Solutions
Digital Educational Services	Mar.	1987	VAX/VMS System Management
Compuserve Data Technologies	Jan.	1987	System 1032 Relational Database
Digital Educational Services	Sep.	1986	VAX/VMS Utilities and Commands
Digital Educational Services	Apr.	1986	VAX/VMS System Performance Management
Digital Educational Services	Nov.	1985	VAX-11 Concepts
AGT Business Systems	Mar.	1985	Networking and Communication Concepts
Software House	Nov.	1984	System 1022 Database Management System
Strathcona County	Sep.	1984	Essentials of Situational Leadership
Digital Educational Services	May	1984	TOPS-20 Monitor Internals
Digital Educational Services	Aug.	1983	Advanced Assembly Language Programming

EMPLOYMENT HISTORY

Coordinator, Planning and Data Services (Jan.82 - Present)

Employed by the County of Strathcona in Sherwood Park, Alberta. The County consists of both a Municipal and Educational component with a combined budget of $120 Million and over 1500 employees. The County's Computing Center has an operating budget of over $1 million and a full time staff of 10 (1 Director, 1 Coordinator, 2 Operators, and 6 Programmer Analysts).

* Directly responsible for all Operations staff, 2 Programmer Analyst and all contract employees. Shared supervision and direction of the other 4 Programmer Analysts with the Director.

* Responsible for micro-to-micro as well as micro-to-mainframe communication (hardware and software acquisition and support).

* Designed, wrote and supported a number of strategic software products and systems:
  - Major portions of the Student Information System for the Board of Education.
  - A computerized registration and results reporting package for the 1987 Alberta Summer Games.
  - A command parser interface for a commercial relational database management system.
  - A computerized time management tracking system.
  - A language sensitive "smart" editor to improve the productivity of the programming staff.
  - All standard system subroutines. These routines are used in the development of all other application subsystems.

* Responsible for the software patches and upgrades to all operating systems, networks and third party software products:
  - Spreadsheets (EXCEL, LOTUS, DECALC, NCPCALC)
  - Networking (Ethernet, DECNET, Pacer, AppleShare, TOPS, TCP/IP)
  - Database Systems (DB1022, DB1032, dBASE, dbRAIMA, HYPERCARD, ORACLE)
  - Statistics (SPSSX, StatWorks)
  - Graphics (MacDraw/Paint, AdobeIllustrator, SPSSGRAPHICS, DECGRAPH, DECSLIDE)
  - Word processing (Word, WordPerfect, WPS+, RUNOFF, SonarProfessional)
  - Document Transfer (DECdx, AFE)
  - Office Automation (ALL-IN-1, INBOX, QuickMail)

* Responsible for Data Center security, communications, space allocation, environment control (power, air, HALON), peripheral equipment acquisition and installation.

* Provided training for our programming staff and user community on the use of both new and existing software packages.

* Responsible for acquisition, support and maintenance of the DEC-Alpha, VAX-8650, MicroVAX-II, DECstations, VAXstations and DECSYSTEM-2060 computer systems plus all associated peripherals.

* Responsible for the development of operational procedures and the production of procedure manuals for the operations staff.

* Shared, with the Data Center Director, responsibility for the following areas:
  - Budget preparation, administration, operation, long range planning.
  - Planning the development, maintenance and scheduling of all production systems: Payroll, Taxation, General Ledger, Accounts Payable, Accounts Receivable, Student Records, Assessment, Utility Billing, Decision Support and Financial Reporting.
  - Contracts (for consulting, software and hardware), furnishings and supplies.
  - Monitoring and supervision of external consultants.

Programmer/Statistical Analyst (Sept.81 - Jan.82)

While a graduate student in computing science, I was hired by D.E.R.S. (Division of Educational Research Services) at the University of Alberta to maintain, modify and create programs for the D.E.R.S. statistical package and to train people on its use. The system ran on the Amdahl mainframe at the University of Alberta.

Teacher - Public School (Sept.79 - Aug.81)

Employed by Birchwood Jr. High School and Dr. K.A. Clark School in Fort McMurray, Alberta. I taught math, biology, physics and computing science to grades seven through nine as well as a specialized program to a group of Trainable Mentally Handicapped students at the elementary level.

Computer Programmer / Statistical Analyst (Jan.79 - Aug.79)

Working at D.E.R.S. (Division of Educational Research Services) for the University of Alberta, I designed, wrote and supported a number of specialized programs for graduate students. Many of these programs were statistical in nature and were used for the analysis of research data.

Water Quality Analyst (May 78 - Sept.78)

Working for the Okanagan Basin Implementation Board, and the Public Health Unit in Penticton, B.C., I collected and analyzed water samples from a number of public

beaches. This summer study was set up to investigate how fecal coliform levels on the beaches varied throughout the summer months

R.C.M.P. Crime Laboratory (May 77 - Sept.77)

As a member of the Serology section in the Crime Laboratory in Edmonton, Alberta I analyzed various body fluids (blood, saliva, vaginal, seminal secretions etc.). Besides the regular case work, research was also done to determine the frequency of certain blood group markers in native populations.

Laboratory Teaching Assistant (Sept.76 - Jan.77)

At the University of Alberta I taught first and second year students the basic laboratory techniques required for their studies in genetics. I also held tutorial sessions to provide additional help to those students having problems with their course material.

Researcher (May 76 - Sept.76)

At the University of Alberta I worked in conjunction with a faculty member of the Department of Genetics on a research project. The project involved determining an ontogeny profile of the X.D.H. enzyme activity for D. melanogaster. This work required a thorough knowledge of genetics as well as such techniques as centrifugation, protein determinations, spectroflorometry, electrophoresis and dialysis.

Researcher (May 75 - Sept.75)

At the University of Alberta I conducted research regarding the feasibility for genetic control of the mosquito A. Vexans. This job required the use of such laboratory techniques as microscopy, microsectioning, x-ray irradiation, and the use of radioactive isotopes. Upon conclusion of the research a paper was written and published (available on request from the Department of Environment).

Recreation Coordinator (May 74 - Sept.74)

This summer job involved assisting in the construction of a recreational program for the younger psychiatric patients at the Penticton Regional Hospital and other selected youths in the Penticton area.

HARDWARE / SOFTWARE

Computing Systems:	DEC-Alpha(RISC), DECstation(RISC), VAX-8650, MicroVAX, VAXstations, Macintosh Family, DECSYSTEM-2060, Amdahl, IBM PC/XT/AT and clones.
Peripheral Equipment:	Disk drives: RZ5x, RA81, RD53, RP07, RP06. Tape drives: TU81+, TU77, TK50. Printers: LP20(Postscript), LN03 (laser), LP14, Apple Laserwriter, HP LaserJet, LG02, LP27, LA210/120/50. Communications: Terminal Servers, 10baseT Hubs, Delni, 513/212/V.32/V.42 modems.
Operating Systems:	OSF/1, VMS, ULTRIX, TOPS-20, A/UX, DOS, Macintosh-OS, MTS
Programming Languages:	C, FORTRAN, LISP, COBOL, MACRO-36 (assembler), PASCAL, APL, ALGOL, VAXTPU (text processing language), PL1022/PL1032 (4th G.L.), Authorware Professional.
Database Management:	System 1022 / 1032 (Relational Database Management System with 4th G.L.), dBASE-III+, dbRAIMA, ORACLE
Spreadsheets:	EXCEL, LOTUS-123, DECalc, NCPCALC
Word processing:	WordPerfect, Microsoft Word, WPS+, RUNOFF
Miscellaneous:	Authorware Professional, SPSSX / SPSSGRAPHICS, DECgraph, DECslide, Microsoft Chart, SCOPE (Screen Formatting Package), ALL-IN-1 (Office Automation Tool), MacPaint, MacDraw, MacProject, Apple Scanner, CD readers, X-windows, OSF Motif.
Networking:	TCP/IP, LAT, DECNET, Ethernet, PACER, Apple Share, Apple Talk, TOPS, Novell, Shiva Telebridge, Kinetic Fastpath, Cayman GatorBox.

PUBLICATIONS

Genetic Control of Aedes Vexans (1976).
Department of Environment, Edmonton, Alberta, Canada.

The Development and Validation of a Computerized IPP Management System through Prototyping (1993). Master Thesis. University of Alberta, Edmonton, Canada.

AWARDS

Best Presentation Software (1993)	Instructional Multi-Media Competition. University of Alberta
Best Multi-Media Software (1993)	Instructional Multi-Media Competition. University of Alberta
Classified Sabbatical (1992/93)	Strathcona County Board of Education Sherwood Park, Alberta.
Provincial Scholarship (1978)	University of Alberta
Queen Elizabeth Scholarship (1976)	University of Alberta
B.C. Medical Association Scholarship (1973)	Penticton, B.C.
B.C. Government Honors Scholarship (1973)	Victoria, B.C.

Diskette Enclosure

The enclosed diskette contains the developed IPP program (Wodelet, 1993) and its associated data files. The diskette was formatted and is readable by all Apple Macintosh computers capable of reading double density (800 KBytes) diskettes. The IPP program was designed using version 7.0 and 7.1 of the Macintosh operating system but has been successfully run on version 6.x systems. The IPP program was written in the Excel macro language (Microsoft, 1992a). It requires a licensed copy of Excel version 3.0 or greater[16]. **Table 33** contains a description of the contents of the enclosed diskette.

---

[16]The IPP program was developed and tested primarily using Excel version 3.0. With the release of Excel 4.0, the program was converted to run under this new version and special code was added to take advantage of some of the new features (such as the automatic setting of portrait or landscape mode when changing formats). The IPP program automatically detects which version of Excel is being used and adjusts its operation accordingly.

Table 33: IPP Diskette Contents

File / Folder	Description
IPP Template	This is the blank template file used by the IPP program to create a new IPP . This is the file you should open (double click on) to create an IPP for a new student
IPP Data Files (Folder)	This is a file folder containing the IPP macros and data files used by the IPP program. Its contents are described below.
IPP Macros	Excel (Microsoft, 1992a) macros used by the IPP program. This macro file is automatically loaded each time the "IPP Template" file (or a student's IPP created from it) is opened. The macros subroutines in it are accessed through buttons on the "IPP Template".
IPP ELA Objectives	Database of standardized objectives used mainly for students in the Enhanced Learning Assistance program.
IPP EMH Objectives	Database of standardized objectives used mainly for students in the Educable Mentally Handicapped program.
IPP TMH Objectives	Database of standardized objectives used mainly for students in the Trainable Mentally Handicapped program.
IPP DH Objectives	Database of standardized objectives used mainly for students in the Dependent Handicapped program.
Sample IPP	A sample IPP created using the IPP program.
IPP User Manual	User manual for the IPP program. This file is in Word (Microsoft, 1992b) version 5.0 format.
Views & Perceptions Database	Excel (Microsoft, 1992a) database used to collect and analyze the users views and perceptions of the developed IPP system (Wodelet, 1993).